(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) **EP 4 060 991 A1**

(12) # EUROPEAN PATENT APPLICATION

(43) Date of publication:
21.09.2022 Bulletin 2022/38

(21) Application number: **22165192.0**

(22) Date of filing: **12.10.2012**

(51) International Patent Classification (IPC):
H04N 19/423 $^{(2014.01)}$ ***H04N 19/44*** $^{(2014.01)}$
H04N 19/573 $^{(2014.01)}$ ***H04N 19/58*** $^{(2014.01)}$
H04N 19/70 $^{(2014.01)}$ ***H04N 19/00*** $^{(2014.01)}$

(52) Cooperative Patent Classification (CPC):
H04N 19/58; H04N 19/423; H04N 19/44; H04N 19/573; H04N 19/70

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR

(30) Priority: **13.10.2011 US 201113273191**
01.11.2011 US 201113287015
08.11.2011 US 201113291961
11.11.2011 US 201113294996

(62) Document number(s) of the earlier application(s) in accordance with Art. 76 EPC:
20194910.4 / 3 767 950
16203741.0 / 3 166 314
12839718.9 / 2 767 090

(71) Applicant: **Dolby International AB**
1097 DP Amsterdam (NL)

(72) Inventors:
- **MISRA, Kiran**
  Camas, 98607 (US)
- **DESHPANDE, Sachi n. G**
  Camas, 98607 (US)
- **SEGALL, Christopher A.**
  Camas, 98607 (US)

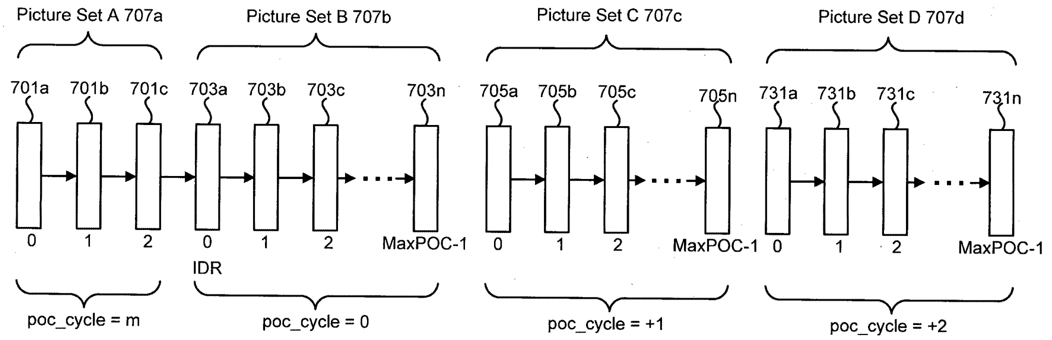
(74) Representative: **Peterreins Schley**
Patent- und Rechtsanwälte PartG mbB
Hermann-Sack-Strasse 3
80331 München (DE)

Remarks:
•This application was filed on 29-03-2022 as a divisional application to the application mentioned under INID code 62.
•Claims filed after the date of filing of the application (Rule 68(4) EPC).

(54) ## TRACKING A REFERENCE PICTURE BASED ON A DESIGNATED PICTURE ON AN ELECTRONIC DEVICE

(57) A method for tracking a reference picture on a decoder is proposed. A modification of buffer description which includes "wrapped POC" and a poc cycle to signal long term pictures is introduced to reduce overhead of referencing pictures.

FIG. 7



Processed by Luminess, 75001 PARIS (FR)

Description

RELATED REFERENCES

[0001] This application is a continuation-in-part of U.S. Patent Application No. 13/291,961 entitled "TRACKING A REFERENCE PICTURE BASED ON A DESIGNATED PICTURE ON AN ELECTRONIC DEVICE," filed on November 8, 2011, which is a continuation-in-part of U.S. Patent Application No. 13/287,015, entitled "TRACKING A REFERENCE PICTURE BASED ON A DESIGNATED PICTURE ON AN ELECTRONIC DEVICE," filed on November 1, 2011, which is a continuation-in-part of U.S. Patent Application No. 13/273,191, entitled "TRACKING A REFERENCE PICTURE ON AN ELECTRONIC DEVICE," filed on October 13, 2011, all of which are hereby incorporated by reference herein, in their entirety.

TECHNICAL FIELD

[0002] The present disclosure relates generally to electronic devices. More specifically, the present disclosure relates to enabling tracking of a reference picture.

BACKGROUND ART

[0003] Electronic devices have become smaller and more powerful in order to meet consumer needs and to Improve portability and convenience. Consumers have become dependent upon electronic devices and have come to expect increased functionality. Some examples of electronic devices include desktop computers, laptop computers, cellular phones, smart phones, media players, integrated circuits, etc.

[0004] Some electronic devices are used for processing and displaying digital media. For example, portable electronic devices now allow for digital media to be consumed at almost any location where a consumer may be. Furthermore, some electronic devices may provide download or streaming of digital media content for the use and enjoyment of a consumer.

[0005] The increasing popularity of digital media has presented several problems. For example, efficiently representing high-quality digital media for storage, transmittal and playback presents several challenges. As can be observed from this discussion, systems and methods that represent digital media more efficiently may be beneficial.

SUMMARY OF INVENTION

[0006] A preferred embodiment is a method for decoding video bitstream comprising: receiving a reference description for a decoded reference pictures from said bitstream; determining a picture order count (POC) of a current picture for decoding; identifying said decoded reference pictures of said current picture from a decoded picture buffer, based on said picture order count and said reference description; decoding said current picture using inter-frame prediction based on said decoded reference pictures; and buffering said decoded picture in said decoded picture buffer to future prediction, wherein said reference description includes a modification parameter for long term reference pictures; and said identifying uses said modification parameter to modify at least a part of references for a set of decoded reference pictures.

[0007] Another preferred embodiment is an electronic device configured for decoding video bitstream, comprising: a processor; memory in electronic communication with the processor; instruction stored in the memory, the instructions being executable to: receive a reference description for a decoded reference pictures from said bitstream; determine a picture order count (POC) of a current picture for decoding; identify said decoded reference pictures of said current picture from a decoded picture buffer, based on said picture order count and said reference description; decode said current pictures using inter-frame prediction based on said decoded reference pictures; and buffer said decoded pictures in said decoded picture buffer to future prediction, wherein said reference description includes a modification parameter for long term reference pictures; and said modification parameter is used to modify at least a part of references for a set of decoded reference pictures.

BRIEF DESCRIPTION OF DRAWINGS

[0008]

Figure 1 is a block diagram illustrating an example of one or more electronic devices in which systems and methods for tracking a reference picture based on a designated picture may be implemented;
Figure 2 is a block diagram illustrating one configuration of a decoder;
Figure 3 is a flow diagram illustrating one configuration of a method for tracking a reference picture with reduced

overhead referencing based on a designated picture;

Figure 4 is a flow diagram illustrating a more specific configuration of a method for tracking a reference picture with reduced overhead referencing based on a designated picture;

Figure 5 is a diagram illustrating one example of multiple picture sets referenced by cycle parameters;

Figure 6 is a diagram illustrating another example of multiple picture sets;

Figure 7 is a diagram of a more specific example of multiple picture sets referenced by cycle parameters;

Figure 8 is a flow diagram illustrating one configuration of a method for tracking a reference picture with reduced overhead referencing based on a designated picture;

Figure 9 is a flow diagram illustrating another configuration of a method for tracking a reference picture with reduced overhead referencing based on a designated picture;

Figure 10 is a flow diagram illustrating another configuration of a method for tracking a reference picture with reduced overhead referencing based on a designated picture;

Figure 11 is a diagram illustrating one example of signaling a wrap indicator in accordance with the systems and methods disclosed herein;

Figure 12 is a diagram illustrating one example of signaling a wrap indicator in accordance with the systems and methods disclosed herein;

Figure 13 is a flow diagram illustrating another more specific configuration of a method for tracking a reference picture with reduced overhead referencing based on a designated picture;

Figure 14 is a flow diagram illustrating one configuration of a method for determining whether a transition has occurred between picture sets;

Figure 15 is a flow diagram illustrating another more specific configuration of a method for tracking a reference picture with reduced overhead referencing based on a designated picture;

Figure 16 illustrates various components that may be utilized in an electronic device; and

Figure 17 is a diagram illustrating examples of use cases in accordance with the systems and methods disclosed herein.

DESCRIPTION OF EMBODIMENTS

[0009] A method for tracking a reference picture on an electronic device is described. The method includes receiving a bitstream. The method also includes decoding a portion of the bitstream to produce a decoded reference picture. The method further includes tracking the decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing based on a designated picture. The method additionally includes decoding a picture based on the decoded reference picture. The designated picture may be an instantaneous decoding refresh (IDR) picture. Additionally, a buffer description of the decoded reference picture may include a picture order count (POC), a cycle parameter, a temporal identifier and a scaling parameter.

[0010] Tracking the decoded reference picture may include determining a cycle parameter based on the designated picture. The cycle parameter may be reset based on the designated picture.

[0011] Tracking the decoded reference picture may include determining a picture order count (POC) based on the designated picture. A picture order count (POC) sequence may be reset based on the designated picture.

[0012] A resolution of the decoded reference picture may be different from a resolution of the picture. The method may also include processing transform coefficients of the decoded reference picture based on a scaling parameter to decode the picture.

[0013] Tracking the decoded reference picture may include tracking a decoded reference picture collection that includes the decoded reference picture. Tracking the decoded reference picture may also include obtaining a buffer description and modifying the buffer description. Modifying the buffer description may include deleting an entry, adding an entry and/or replacing an entry.

[0014] An electronic device configured for tracking a reference picture is also described. The electronic device includes a processor and instructions stored in memory that is in electronic communication with the processor. The electronic device receives a bitstream. The electronic device also decodes a portion of the bitstream to produce a decoded reference picture. The electronic device further tracks the decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing based on a designated picture. The electronic device additionally decodes a picture based on the decoded reference picture.

[0015] The systems and methods disclosed herein describe several configurations for tracking a reference picture based on a designated picture on an electronic device. For example, the systems and methods disclosed herein describe tracking a decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing. For instance, several approaches for long term reference picture signaling are described. It should be noted that the decoded picture buffer (DPB) may be a buffer holding decoded pictures for reference, output reordering or output delay specified for a hypothetical reference decoder.

[0016] On an electronic device, a decoded picture buffer (DPB) may be used to store reconstructed (e.g., decoded) pictures at a decoder. These stored pictures may then be used, for example, in an inter-prediction mechanism. When pictures are decoded out of order, the pictures may be stored in the DPB so they can be displayed later in order.

[0017] In the H.264 or advance video coding (AVC) standard, DPB management (e.g., deletion, addition of pictures, reordering of pictures, etc.) is carried out using memory management control operations (MMCO). For the upcoming high efficiency video coding (HEVC) standard, more reliable DPB management approaches are under consideration. One example of a more reliable approach is based on absolute signaling of reference pictures as detailed in "Absolute signaling of reference pictures" from the Joint Collaborative Team on Video Coding (JCT-VC) document JCTVC-F493.

[0018] JCTVC-F493 outlines absolute signaling of reference pictures to identify which reference pictures should be kept in the decoded picture buffer (DPB). In particular, it outlines two different approaches to identify which reference pictures are to be kept in the DPB based on a picture order count (POC). The picture order count (POC) may be a variable that is associated with each encoded picture and has a value that is increasing with increasing picture position in an output order with wrap-around.

[0019] In one example, assume that all pictures have a temporal identifier (temporalID) = 0. Further assume that the current POC = 5 and that the current DPB contains = {3, 2}. Additionally assume that a definition in the Picture Parameter Set (PPS) is: BufferDescription0 = {deltaPOC = -1, temporalID = 0}, {deltaPOC = -2, temporalID = 0}. deltaPOC specifies the distance in POC value of reference picture from the current picture, where the current picture may be the picture being decoded. One approach given is to reference a buffer description in the PPS. In this approach, the slice header of a picture with POC = 5 contains a reference to BufferDescription0 in the PPS. Assume that an action is to drop a decoded picture with POC = 2 from the DPB and to add a decoded picture with POC = 4 to the DPB. As a result, the DPB then contains = {4, 3}.

[0020] In one configuration, the buffer description is defined as two lists denoted POCBD and TemporalIDBD and a variable NumberOfPicturesInBD present for all pictures. This is such that POCBD contains picture order count values of reference pictures and TemporalIDBD contains the corresponding temporal identifier of the reference picture, both lists containing NumberOfPicturesInBD values.

[0021] It should be noted that a temporalID may be defined as follows in the Joint Collaborative Team on Video Coding (JCT-VC) document JCTVC-F803: "temporalID specifies a temporal identifier for the NAL unit. The value of temporalID shall be the same for all NAL units of an access unit. When an access unit contains any NAL unit with the nal_unit_type equal to 5, temporalID shall be equal to 0." It should be noted that NAL may be an abbreviation for "network abstraction layer."

[0022] Another approach is to explicitly signal the contents of the DPB using a delta POC with respect to the current POC. In this approach, the slice header of a picture with POC = 5 contains {deltaPOC = -1, temporalID = 0} and {deltaPOC = -2, temporalID = 0}. Assume that an action is to drop a decoded picture with POC = 2 from the DPB and to add a decoded picture POC = 4 to the DPB. As a result, the new DPB contains = {4, 3}.

[0023] Some advantages of the approaches given by JCTVC-F493 are as follows. The approaches in JCTVC-F493 provide a simple mechanism. Furthermore, a loss of a picture is easily detected at the decoder. Additionally, dropping of entire layers of pictures with a higher temporal ID may be detected and well supported.

[0024] However, some disadvantages of the approaches given in JCTVC-F493 are given hereafter. The bit overhead for signaling a long-term reference picture can become large. Furthermore, a fixed number of bits may be allocated to communicate a POC. As a result, when a maximum value allowed by the number of bits being used is reached, the POC numbering should wrap around to 0. Thus, it may not be possible to guarantee that pictures can be uniquely identified using the POC.

[0025] The systems and methods disclosed herein may help to mitigate these disadvantages. In particular, the systems and methods disclosed herein may be beneficial by reducing the overhead associated with absolute long term picture referencing and may enable pictures to be uniquely identified (e.g., a long-term (reference) picture may not be confused with other short-term or long-term pictures and vice-versa).

[0026] The systems and methods disclosed herein may provide one or more additional benefits that are described as follows. One or more configurations of the systems and methods disclosed herein may make full use of the available POC numbering space $[0, \cdots, MaxPOC -1]$, where $MaxPOC = 2^{\mathrm{log2\_max\_pic\_order\_cnt\_minus4+4}}$ and log2_max_pic_order_cnt_minus4 specifies the value of the variable *MaxPOC* that is used in the decoding process for picture order count. For example, one prior approach to resolving re-use of $[0, \cdots, MaxPOC -1]$ after a POC wrap-around advocates that the POC currently in use are stepped over when assigning an identifier (e.g., a POC number) to a picture. This results in part of the POC space not being used. However, the systems and methods disclosed herein may resolve the stepping over of POC and the associated POC space shrinkage issue.

[0027] Another benefit may be that some configurations of the systems and methods disclosed herein for signaling may be self-contained in each picture. Thus, error resilience may be better compared to a scheme that relies on information propagation from previous pictures (that could get lost or dropped). For example, one configuration of the decoded picture buffer (DPB) description does not rely on information embedded in other pictures to maintain the same DPB as an encoder.

[0028] Yet another benefit of some configurations of the systems and methods disclosed herein may be that if a picture is lost, the loss can be detected as soon as a buffer description is available at the decoder (which is at the next received picture). This allows the decoder to take corrective action. Yet another benefit is that if the POC resolution is sufficient, no extra bits are required.

[0029] Various configurations are now described with reference to the Figures, where like reference numbers may indicate functionally similar elements. The systems and methods as generally described and illustrated in the Figures herein could be arranged and designed in a wide variety of different configurations. Thus, the following more detailed description of several configurations, as represented in the Figures, is not intended to limit scope, as claimed, but is merely representative of the systems and methods.

[0030] Figure 1 is a block diagram illustrating an example of one or more electronic devices 104 in which systems and methods for tracking a reference picture based on a designated picture may be implemented. In this example, electronic device A 104a and electronic device B 104b are illustrated. However, it should be noted that the features and/or functionality described in relation to electronic device A 104a and electronic device B 104b may be combined into a single electronic device in some configurations.

[0031] Electronic device A 104a includes an encoder 108 and an overhead signaling module 112. Each of the elements included within electronic device A 104a (e.g., the encoder 108 and the overhead signaling module 112) may be implemented in hardware, software or a combination of both.

[0032] Electronic device A 104a may obtain an input picture 106. In some configurations, the input picture 106 may be captured on electronic device A 104a using an image sensor, retrieved from memory and/or received from another electronic device.

[0033] The encoder 108 may encode the input picture 106 to produce encoded data 110. For example, the encoder 108 may encode a series of input pictures 106 (e.g., video). In one configuration, the encoder 108 may be a high-efficiency video coding (HEVC) encoder. The encoded data 110 may be digital data (e.g., a bitstream).

[0034] The overhead signaling module 112 may generate overhead signaling based on the encoded data 110. For example, the overhead signaling module 112 may add overhead data to the encoded data 110 such as slice header information, picture parameter set (PPS) information, picture order count (POC), reference picture designation, etc. In some configurations, the overhead signaling module 112 may produce a wrap indicator that indicates a transition between two sets of pictures.

[0035] More detail on kinds of overhead signaling that may be produced by electronic device A 104a is given below. In particular, none, one or more of the parameters, indicators or kinds of information described in relation to decoding below may be produced by the overhead signaling module 112, depending on the configuration. It should be noted that the overhead signaling module 112 may be included within the encoder 108 in some configurations. The overhead signaling module 112 may enable picture tracking with reduced overhead referencing.

[0036] The encoder 108 (and overhead signaling module 112, for example) may produce a bitstream 114. The bitstream 114 may include encoded picture data based on the input picture 106. In some configurations, the bitstream 114 may also include overhead data, such as slice header information, PPS information, etc. More detail on overhead data is given below. As additional input pictures 106 are encoded, the bitstream 114 may include one or more encoded pictures. For instance, the bitstream 114 may include one or more encoded reference pictures and/or other pictures.

[0037] The bitstream 114 may be provided to a decoder 102. In one example, the bitstream 114 may be transmitted to electronic device B 104b using a wired or wireless link. In some cases, this may be done over a network, such as the Internet or a Local Area Network (LAN). As illustrated in Figure 1, the decoder 102 may be implemented on electronic device B 104b separately from the encoder 108 on electronic device A 104a. However, it should be noted that the encoder 108 and decoder 102 may be implemented on the same electronic device in some configurations. In an implementation where the encoder 108 and decoder 102 are implemented on the same electronic device, for instance, the bitstream 114 may be provided over a bus to the decoder 102 or stored in memory for retrieval by the decoder 102.

[0038] The decoder 102 may be implemented in hardware, software or a combination of both. In one configuration, the decoder 102 may be a high-efficiency video coding (HEVC) decoder. The decoder 102 may receive (e.g., obtain) the bitstream 114. The decoder 102 may generate a decoded picture 118 (e.g., one or more decoded pictures 118) based on the bitstream 114. The decoded picture 118 may be displayed, played back, stored in memory and/or transmitted to another device, etc.

[0039] The decoder 102 may include a reference picture tracking module 116. The reference picture tracking module 116 may enable the decoder 102 to track a reference picture with reduced overhead referencing. For example, the reference picture tracking module 116 may track a reference picture in a decoded picture buffer (DPB) using less overhead than is needed with prior approaches, such as approaches given in JCTVC-F493.

[0040] In prior approaches, for example, non-reduced overhead referencing may be used to specify the relationship between a current picture and a long term reference picture. In prior approaches, for instance, the relationship between a current picture and a long term reference picture may be specified by increasing the POC numbering space and thereby avoiding the POC wraparound issue. However, increasing the POC numbering space can only be achieved at the expense of an increased bit-requirement for POC. This example is one of several possible mechanisms that can be used to avoid the POC wrap around issue in prior approaches. However, this particular example demonstrates the larger overhead aspect for long-term pictures in prior approaches.

[0041] JCTVC-F493, for example, used a longterm_poc[i] field in a buffer description that specified an absolute POC and a longterm_temporal_id[i] field in the buffer description that specified a temporal ID for a long term picture. This was later removed in JCTVC-F803, which did not include a mechanism for long term pictures. In subsequent discussions, an approach of stepping over (long term picture) POCs was given.

[0042] Problems may arise with the prior approaches. First, a large amount of overhead data may be needed to specify the relationship between a long term reference picture and another picture. For instance, a large number of overhead bits may need to be allocated to adequately represent an integer number difference in POC between the long term reference picture and another picture. Second, if a limited number of bits is specified to represent this difference, the difference may be ambiguously indicated when numbers are reused (because of number set cycling, for example).

[0043] The reference picture tracking module 116 may use one or more approaches or methods that are described in greater detail below in order to reduce referencing overhead. Some examples include using a cycle parameter and decrementing the cycle parameter based on wrap indicators or transitions between sets of pictures.

[0044] It should be noted that a "reference picture" described herein may instead be a reference picture collection (e.g., group of frames) in some configurations. Accordingly, a "decoded reference picture" described herein may instead be a "decoded reference picture collection" in some configurations. For example, a reference picture collection may be applied in place of a single reference picture as described in conjunction with the systems and methods disclosed herein. The reference picture collection may include one or more reference pictures. Two or more reference pictures in the reference picture collection may correspond to the same time instant or different (e.g., similar) time instances. For example, in a 3-dimensional (3D) coding scenario, a bitstream 114 contains multiple pictures, where some pictures may refer to a left view and some pictures may refer to a right view. Thus, the reference picture collection may identify both the left and right pictures that correspond to a specific display time.

[0045] In another example, such as a scalable coding scenario, a bitstream may contain pictures of different resolutions. In this example, the reference picture collection may include (e.g., identify) different resolution versions of the same picture.

[0046] Figure 2 is a block diagram illustrating one configuration of a decoder 202. The decoder 202 may be included in an electronic device 204. For example, the decoder 202 may be a high-efficiency video coding (HEVC) decoder. The decoder 202 and/or one or more of the elements illustrated as included in the decoder 202 may be implemented in hardware, software or a combination of both. The decoder 202 may receive a bitstream 214 (e.g., one or more encoded pictures included in the bitstream 214) for decoding. In some configurations, the received bitstream 214 may include received overhead information, such as a received slice header, received PPS, received buffer description information, etc. The encoded pictures included in the bitstream 214 may include one or more encoded reference pictures and/or one or more other encoded pictures.

[0047] Received symbols (in the one or more encoded pictures included in the bitstream 214) may be entropy decoded by an entropy decoding module 254, thereby producing a motion information signal 256 and quantized, scaled and/or transformed coefficients 258.

[0048] The motion information signal 256 may be combined with a portion of a reference frame signal 284 from a frame memory 264 at a motion compensation module 260, which may produce an inter-frame prediction signal 268. The quantized, descaled and/or transformed coefficients 258 may be inverse quantized, scaled and inverse transformed by an inverse module 262, thereby producing a decoded residual signal 270. The decoded residual signal 270 may be added to a prediction signal 278 to produce a combined signal 272. The prediction signal 278 may be a signal selected from either the inter-frame prediction signal 268 or an intra-frame prediction signal 276 produced by an intra-frame prediction module 274. In some configurations, this signal selection may be based on (e.g., controlled by) the bitstream 214.

[0049] The intra-frame prediction signal 276 may be predicted from previously decoded information from the combined signal 272 (in the current frame, for example). The combined signal 272 may also be filtered by a de-blocking filter 280. The resulting filtered signal 282 may be written to frame memory 264. The resulting filtered signal 282 may include a decoded picture.

[0050] The frame memory 264 may include a decoded picture buffer (DPB) as described herein. The DPB may include one or more decoded pictures that may be maintained as short or long term reference frames. The frame memory 264 may also include overhead information corresponding to the decoded pictures. For example, the frame memory 264 may include slice headers, picture parameter set (PPS) information, cycle parameters, buffer description information, etc. One or more of these pieces of information may be signaled from an encoder (e.g., encoder 108, overhead signaling module 112). The frame memory 264 may provide a decoded picture 218.

[0051] The decoder 202 may include a reference picture tracking module 216. The reference picture tracking module

216 may track one or more reference pictures in the frame memory 264 with reduced referencing overhead. In one example, the reference picture tracking module 216 may track long term reference pictures using a cycle parameter and modifying (e.g., decrementing) the cycle parameter based on received wrap indicators. In another example, an update of all reference picture cycle parameters may be carried out with respect to the picture being decoded. This update procedure may be executed once for the current picture (e.g., the picture being decoded). The transition between cycles may be kept track of implicitly with the help of the POC. At times the cycle parameter may be increased (when the transition is from picture set 'n' to picture set 'n-1' as may occur in out-of-order picture decoding, for example). Greater detail on one or more approaches to tracking a reference picture based on a designated picture is given below.

[0052] Some configurations of the systems and methods disclosed herein may use a modified buffer description. Examples of the modified buffer description are given hereafter. The buffer description may be modified to include "POC," "poc_cycle" and "temporalID" for long-term reference pictures. It should be noted that "poc_cycle" may be one example of the cycle parameter described herein. The (modified) buffer descriptions, (modified) syntaxes and/or parameters given in accordance with the systems and methods disclosed herein may enable reduced overhead referencing.

[0053] Table (1) below gives one example comparing the buffer description within the PPS in a prior approach and one proposed approach in accordance with the systems and methods disclosed herein. The prior approach is detailed in the "candidate working draft text of ad-hoc group 21" document (AHG21) that was created to further the work in JCTVC-F493. It should be noted that AHG21 (JCTVC-F803) separately groups and specifies "negative pictures" (e.g., those that have negative deltaPOC values) and "positive pictures" (e.g., those pictures that have positive deltaPOC values).

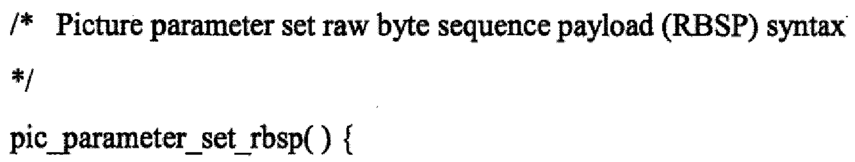
Table (1)

AHG21 Buffer Description	Proposed Buffer Description
$deltaPOC_0$, $temporalID_0$	$deltaPOC_0$, $temporalID_0$
$deltaPOC_1$, $temporalID_1$	$deltaPOC_1$, $temporalID_1$
$deltaPOC_2$, $temporalID_2$	$deltaPOC_2$, $temporalID_2$
$deltaPOC_3$, $temporalID_3$	(POC_0, poc_cycles_0, $temporalID_3$)
$deltaPOC_4$, $temporalID_4$	(POC_1, poc_cycles_1, $temporalID_4$)

[0054] In Table (1) illustrated above, (POC_0, poc_cycles_0, $temporalID_3$) and (POC_1, poc_cycles_1, $temporalID_4$) represent long-term (reference) pictures. It should be noted that the buffer description may contain two lists POCBD and TemporalIDBD for short-term reference pictures (corresponding to POC and TemporalID fields, respectively). Furthermore, the buffer description may contain three lists: POCBD, POC_CYCLE_BD and TemporalIDBD for long-term reference pictures (corresponding to POC, poc_cycle and TemporalID fields, respectively).

[0055] In some configurations, multiple buffer descriptions are defined in the picture parameter set. Each buffer description, with index i, contains lists for reference pictures: POCBD_pps[i], POC_CYCLE_pps[i], DeltaPOCBD_pps[i] and TemporalIDBD_pps[i] and a variable NumberOfPicturesInBD_pps[i]. The i-th list of POCBD_pps[i] contains picture order count values of reference pictures. The i-th list of POC_CYCLE_pps[i] contains poc_cycle values of reference pictures. The i-th list of TemporalIDBD_pps[i] contains the corresponding temporal identifier of the reference picture. Furthermore, the i-th list of TemporalIDBD_pps[i] contains NumberOfPicturesInBD_pps[i] entries. The collection of lists are referred to as POC_pps, POC_CYCLE_pps, DeltaPOCBD_pps and TemporalIDBD_pps. For deltaPOC referencing, a single list DeltaPOCBD_pps[i] contains the deltaPOC values of reference pictures. It should be noted that in the candidate working draft text of ad-hoc group 21, DeltaPOCBD_pps may be referred to as DeltaPOCBD. The sum of the entries in the list POCBD_pps[i] and the list DeltaPOCBD_pps[i] is given by the value NumberOfPicturesInBD_pps[i]. Moreover, the number of entries in POCBD_pps[i] and POC_CYCLE_pps[i] are the same.

[0056] It should be noted that the syntax given in AHG21 does not adequately support fixed long term referencing. Listing (1) below illustrates one example of a bitstream syntax modification required to a candidate working draft text of ad-hoc group 21 (AHG21). The changes due to the prior approach are given in bold text in Listing (1).



```
/* Picture parameter set raw byte sequence payload (RBSP) syntax
*/
pic_parameter_set_rbsp( ) {
```

```
…
bits_for_temporal_id_in_buffer_descriptions
positive_pictures_in_buffer_descriptions_flag
number_of_bds
if( number_of_bds > 0 ) {
    for(i = 0; i < number_of_bds; i++){
        number_of_negative_pictures_pps[i]
        for( j = 0; j < number_of_negative_pictures_pps[i]; j++ ) {
            negative_delta_poc_minus_one_pps[i][j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
            temporal_id_negative_pps[i][j]

        }
        if( positive_pictures_in_buffer_descriptions_flag ){
            number_of_positive_pictures_pps[i]
            for( j = 0; j < number_of_positive_pictures_pps[i]; j++ ) {
                delta_poc_minus_one_pps[i][j]
                if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                    temporal_id_positive_pps[i][j]
            }
        }
        number_of_longterm_pictures_pps[i]
        for( j = 0; j < number_of_longterm_pictures_pps[i]; j++ ) {
            poc_pps[i][j]
            poc_cycle_pps[i][j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_poc_pps[i][j]
        }
    }
}
…
}
```

Listing (1)

[0057] positive_pictures_in_buffer_descriptions_flag specifies if there are any buffer description pictures with positive deltaPOC. bits_for_temporal_id_in_buffer_descriptions specifies the number of bits used for temporal_id_negative_pps[i][j], temporai_id_positive_pps[i][j], temporal_id_poc_pps[i][j], temporal_id_negative[i] and temporal_id_positive[i]. number_of_bds specifies the number of buffer descriptions in a picture parameter set. number_of_negative_pictures_pps[i] specifies the number of entries in the list DeltaPOCBD_pps[i] with negative values. number_of_negative_pictures_pps[i] defines the value of OffsetBD[i] as OffsetBD[i] = number_of_negative_pictures_pps[i]. The value of number_of_negative_pictures_pps[i] shall be in the range of 0 to max_num_ref_frames, inclusive. negative_delta_poc_minus_one_pps[i][j] specifies an absolute distance in POC value. max_num_ref_frames specifies the maximum number of reference frames, complementary reference field pairs, and non-paired reference fields that may be used by the decoding process for inter prediction of any picture in the sequence. Here, a reference field denotes a portion of a reference frame. For example, in an application using interlaced video, a reference frame may consist of two reference fields. The first reference field may contain a first subset of the data in the reference frame and the second reference field may contain a second subset of the data in the reference frame, where the first subset and second subset correspond to different data.

[0058] negative_delta_poc_minus_one_pps[i][j] defines the value of the variable DeltaPOCBD_pps[i][j] as DeltaPOCBD_pps[i][j] = - (negative_delta_poc_minus_one_pps[i][j] + 1) if j equals 0. Also, negative_delta_poc_minus_one_pps[i][j] defines the value of the variable DeltaPOCBD_pps[i][j] as DeltaPOCBD_pps[i][j] = DeltaPOCBD_pps[i][j-1] - (negative_delta_poc_minus_one_pps[i][j] + 1) if j > 0. DeltaPOCBD_pps[i][j] shall be in the range of -1 to -MaxPOC/2.

[0059] temporal_id_negative_pps[i][j] specifies a temporal identifier and shall be represented by bits_for_temporal_id_in_buffer_descriptions bits. temporal_id_negative_pps[i][j] is added to the list TemporalIDBD_pps[i]. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending operation is an operation that replaces items in the list in a predefined order. temporal_id_negative_pps[i][j] shall be in the range of 0 to max_temporal_layers_minus1, inclusive. max_temporal_layers_minus1+1 specifies the maximum number of temporal layers present in a sequence.

[0060] number_of_positive_pictures_in_bd_pps[i] specifies the number of entries in the list DeltaPOCBD_pps[i] that have a positive value. The value of the number_of_positive_pictures_in_bd_pps[i] shall be in the range of 0 to max_num_ref_frames-OffsetBD[i], inclusive. When the number_of_positive_pictures_in_bd_pps[i] syntax element is not present, the value of number_of_positive_pictures_in_bd_pps[i] shall be inferred to be equal to 0.

[0061] delta_poc_minus_one_pps[i][j] specifies an absolute distance in POC value. delta_poc_minus_one_pps[i][j] defines the value of the variable DeltaPOCBD_pps[i][j+OffsetBD[i]] as DeltaPOCBD_pps[i][j+OffsetBD[i]] = delta_poc_minus_one_pps[i][j] + 1 if j is equal to 0 and as DeltaPOCBD_pps[i][j+OffsetBD[i]] = DeltaPOCBD_pps[i][j-1+OffsetBD[i]] + delta_poc_minus_one_pps[i][j] + 1 if j > 0. DeltaPOCBD_pps[i][j+OffsetBD[i]] shall be in the range of 1 to MaxPOC/2-1.

[0062] temporal_id_positive_pps[i][j] specifies a temporal identifier and shall be represented by bits_for_temporal_id_in_buffer_descriptions bits. temporal_id_positive_pps[i][j] defines the value of the variable TemporalIDBD_pps[i][j] as TemporalIDBD_pps[i][j+OffsetBD[i]] = temporal_id_positive_pps[i][j]. temporal_id_positive_pps[i][j] shall be in the range of 0 to max_temporal_layers_minus1, inclusive.

[0063] Examples of descriptions of the parameters in Listing (1) are given as follows. number_of_longterm_pictures_pps[i] specifies the number of entries in the list POCBD_pps[i] and POC_CYCLE_BD_pps[i]. The value of number_of_longterm_pictures_pps[i] shall be in the range of 0 to max_num_ref_frames, inclusive. max_num_ref_frames specifies the maximum number of short term and long term reference frames. poc_pps[i][j] specifies POC value and defines the value to be added to the list POCBD_pps[i] as POCBD[i][j] = poc[j]. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order. poc_pps[i][j] shall be in the range of 0 to MaxPOC-1.

[0064] poc_cycle_pps[i][j] specifies poc_cycle (e.g., the cycle parameter) value and defines the value to be added to the list POC_CYCLE_BD_pps[i]. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order. poc_cycle_pps[i][j] (e.g., the cycle parameter) may be less than or equal to zero in some configurations. In such a case, a signed integer may be used to represent the cycle parameter. In other configurations, an unsigned integer may be used to represent the cycle parameter.

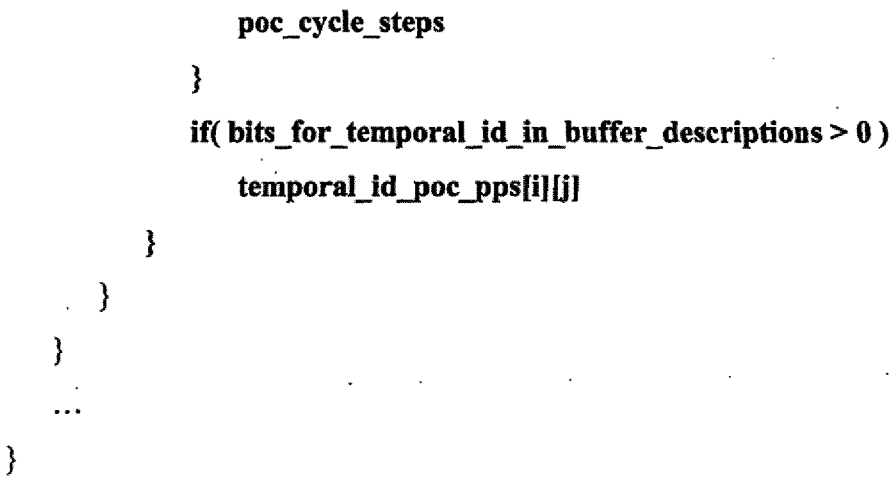
[0065] temporal_id_poc_pps[i][j] specifies a temporal identifier and shall be present if bits_for_temporal_id_in_buffer_descriptions > 0. temporal_id_poc_pps[i][j] defines the value of the entries in the list

TemporalIDBD_pps[i]. temporal_id_poc_pps[i][j] is added to the list TemporalIDBD_pps[i]. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order. temporal_id_poc_pps[i][j] shall be in the range of 0 to max_temporal_layers_minus1, inclusive. max_temporal_layers_minus1+1 specifies the maximum number of temporal layers present in a sequence. It should be noted that the variable NumberOfPicturesInBD_pps[i] may be calculated as number_of_negative_pictures_pps[i] + number_of_positive_pictures_pps[i] + number_of_longterm_pictures_pps[i].

[0066] The negative and positive deltaPOC values for the reference picture transmitted in the i-th picture parameter set buffer description are added to the list DeltaPOCBD_pps[i]. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order.

[0067] Listing (2) below illustrates an alternative example configuration where multiple buffer descriptions may be created within a PPS with different cycle parameters (e.g., poc_cycles) using the following syntax. The changes due to the prior approach are given in bold text in Listing (2).

```
/* Picture parameter set RBSP syntax
*/
pic_parameter_set_rbsp( ) {
    ...
    bits_for_temporal_id_in_buffer_descriptions
    positive_pictures_in_buffer_descriptions_flag
    number_of_bds
    if( number_of_bds > 0 ) {
        for(i = 0; i < number_of_bds; i++){
            number_of_negative_pictures_pps[i]
            for( j = 0; j < number_of_negative_pictures_pps[i]; j++ ) {
                negative_delta_poc_minus_one_pps[i][j]
                if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_negative_pps[i][j]
            }
            if( positive_pictures_in_buffer_descriptions_flag ){
                number_of_positive_pictures_pps[i]
                for( j = 0; j < number_of_positive_pictures_pps[i]; j++ ) {
                    delta_poc_minus_one_pps[i][j]
                    if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                        temporal_id_positive_pps[i][j]
                }
            }
            number_of_longterm_pictures_pps[i]
            for( j = 0; j < number_of_longterm_pictures_pps[i]; j++ ) {
                poc_pps[i][j]
                poc_cycle_pps[i][j]
                poc_cycle_steps_flag
                if (poc_cycle_steps_flag) {
```



```
                    poc_cycle_steps
                }
                if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                    temporal_id_poc_pps[i][j]
            }
        }
    }
    ...
}
```

Listing (2)

[0068] In Listing (2), examples of descriptions of further parameters are given as follows. When set to 1, poc_cycle_steps_flag specifies that additional buffer descriptions shall be generated for the signaled buffer description model that are identical to the signaled buffer description model except for the poc_cycle count. poc_cycle_steps_flag shall be 0 by default. Furthermore, poc_cycle_steps specifies the number of additional buffer descriptions that shall be generated for the signaled buffer description model. The additional buffer descriptions shall be identical to the signaled buffer description except that the poc_cycle count shall be decreased. In one configuration, the additional buffer descriptions generated have poc_cycle_pps[i][j] values of -1 ,-2,-3,...,-(poc_cycle_steps).

[0069] In some configurations, resolution switching may be enabled within a bitstream. In these configurations, the resolution of the reference pictures may be different from the resolution of the current picture. Thus, it may be beneficial to know which mechanism may be used to scale the reference pictures to the right resolution. One way of signaling the mechanism is to signal it explicitly along with the buffer description. For example, an encoder 108 (e.g., overhead signaling module 112) may signal a scaling parameter *s* to the decoder 102. Thus, an alternative buffer description may be shown in Table (2).

Table (2)

Alternative Buffer Description
$deltaPOC_0$, $temporalID_0$
$deltaPOC_1$, $temporalID_1$
$deltaPOC_2$, $temporalID_2$, s_0
(POC_0, poc_cycles_0, $temporalID_3$,s_1)
(POC_1, poc_cycles_1, $temporalID_4$,s_2)

[0070] In Table (2), the first two entries have the resolution matching that of the current picture. However the remaining three entries have a different resolution and may use the scaling parameter, s_0, s_1. s_2, respectively. The scaling parameter may be signaled selectively. For example, it may be signaled only if the current picture and the reference picture have a different resolution. In another configuration, the scaling parameter may always be signaled. In yet another configuration the scaling parameter may be inferred implicitly (by a decoder), for example, by using a look up table indexed by current picture characteristics (e.g., resolution) and the referenced picture characteristics (e.g., resolution).

[0071] In yet another configuration, the scaling parameter may be signaled for any subset of pictures defined in the buffer description. For example, the scaling parameter may be signaled for a subset of reference pictures signaled using positive deltaPOC values, a subset of reference pictures signaled using negative deltaPOC values, a subset of pictures signaled using POC and poc_cycle. In one configuration, a subset may contain the entire list. One example is illustrated in Table (3) below, where $negativeDeltaPOC_0$ and $negativeDeltaPOC_1$ are negative delta POC values, $positiveDeltaPOC_2$ is a positive delta POC value and s_0, s_1, s_2, s_3 and s_4 are the scaling parameters for a first entry, second entry, third entry, fourth entry and fifth entry, respectively.

Table (3)

Alternative Buffer Description
$\mathit{negativeDeltaPOC}_0$, $\mathit{temporalID}_0$,s_0
$\mathit{negativeDeltaPOC}_1$, $\mathit{temporalID}_1$,s_1
$\mathit{positiveDeltaPOC}_2$, $\mathit{temporalID}_2$,s_2
(POC_0, $\mathit{poc\_cycles}_0$, $\mathit{temporalID}_3$,s_3)
(POC_1, $\mathit{poc\_cycles}_1$, $\mathit{temporalID}_4$, s_4)

[0072] In yet another configuration, the scaling parameter may indicate the one or more resolutions of the reference picture to be kept in the decoded picture buffer. In yet another configuration, the scaling parameter may be used to signal the resolution of the reference picture to be used for a motion compensation process. In yet another configuration, if the resolution of the reference picture and the picture being decoded are not the same, the scaling parameter indicates the motion compensation process to use to generate predictions.

[0073] In one configuration, the scaling parameter signals (e.g., indicates) the change in horizontal and/or vertical resolution in pixels. In one configuration, the scaling parameter signals (e.g., indicates) the ratio between the desired and original horizontal and/or vertical resolution. In one configuration, the scaling parameter is a 2-tuple, where the first value identifies scaling for the horizontal resolution and the second value identifies scaling for the vertical resolution.

[0074] In some configurations, the transform coefficients of the reference picture may be processed based on the scaling parameter to obtain the desired resolution. This may be beneficial to enable better resolution adaptation by allowing only block sizes larger than a certain threshold while coding the reference picture.

[0075] In yet another configuration, the reconstructed pixels of the reference picture may be processed based on the scaling parameter to obtain the desired resolution. It should be noted that these configurations may be applied to all reference picture indexing schemes disclosed herein.

[0076] Listing (3) illustrates another example of syntax modification for the PPS from AHG21. In particular, Listing (3) illustrates one example of buffer description syntax used in slice headers as outlined in AHG21. However, modifications to the syntax given in AHG21 in accordance with the systems and methods disclosed herein are denoted in bold text in Listing (3).

```
/*  Buffer description
*/
buffer_description( ) {
    bd_reference_flag
    if(bd_reference_flag = = 1) {
        bd_idx
        bd_poc_cycle_update_flag
        if (bd_poc_cycle_update_flag == 1)
            //begin for loop of long term pictures POCBD_pps[bd_idx],
            POC_CYCLE_BD_pps[bd_idx] and
            TemporalIDBD_pps[bd_idx]override
            for( j = 0; j < number_of_longterm_pictures_pps[bd_idx]; j++ ) {
                poc_pps_override[j]
                poc_cycle_pps_override[j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_pps_override[j]
        }
        //end of for loop of long term pictures POCBD_pps[bd_idx],
        POC_CYCLE_BD_pps[bd_idx] and
        TemporalIDBD_pps[bd_idx]override
} else {
    number_of_negative_pictures
    //begin for loop for negative deltaPOC pictures
    for( i = 0; i < number_of_negative_pictures; i++ ) {
        negative_delta_poc_minus_one[i]
        if( bits_for_temporal_id_in_buffer_descriptions > 0 )
            temporal_id_negative[i]                         }
    // end of for loop for negative deltaPOC pictures
    if( positive_pictures_in_buffer_descriptions_flag ){
        delta_poc_minus_one[i]
        if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_positive[i]             }
    number_of_longterm_pictures
    // begin for loop of explicitly signaled long term pictures
    for( j = 0; j < number_of_longterm_pictures; j++ ) {
        poc[j]
        poc_cycle[j]
        if( bits_for_temporal_id_in_buffer_descriptions > 0 )
            temporal_id_poc[j]
    }
    //end of for loop of explicitly signaled long term pictures
    if( number_of_negative_pictures + number_of_positive_pictures <
         max_num_ref_frames ) {
        combine_with_reference_flag
        if( combine_with_reference_flag )
            bd_combination_idx
        }
    }
}
```

Listing (3)

[0077] Examples of descriptions of the parameters in Listing (3) are given as follows. A number_of_negative_pictures specifies the number of negative delta POC entries. number_of_negative_pictures defines the value of Offset as Offset = number_of_negative_pictures. The value of number_of_negative_pictures shall be in the range of 0 to max_num_ref_frames, inclusive.

[0078] negative_delta_poc_minus_one[i] specifies an absolute distance In POC value. negative_delta_poc_minus_one[i] defines the value of the variable POCBD[i] as POCBD[i] = (pic_order_cnt + MaxPOC - (negative_delta_poc_minus_one[i] + 1)) % MaxPOC if i equals 0 and as POCBD[i] = (POCBD[i-1] + MaxPOC - (negative_delta_poc_minus_one[i] + 1)) % MaxPOC if i > 0. Here, pic_order_cnt is the POC of the current picture as signaled in the slice header. The value of negative_delta_poc_Minus_one[i] shall be in the range of 0 to MaxPOC-1, inclusive. The value of POCBD[i] shall be such that the value of DiffPOC(currPic, refPic) is in the range of -1 to -MaxPOC/2. Here, refPic is the reference picture with pic_order_cnt equal to POCBD[i].

[0079] DiffPOC(picA, picB) is specified as follows:

```
if( POC(picA) > POC(picB) ) {
           if(POC(picA) - POC(picB) < MaxPOC / 2 )
                 DiffPOC(picA, picB) = POC(picA) - POC(picB)
           else
                 DiffPOC(picA, picB) = POC(picA) - MaxPOC - POC(picB)
      else {
           if(POC(picB) - POC(picA) <= MaxPOC / 2 )
                 DiffPOC(picA, picB) = POC(picA) - POC(picB)
           else
                 DiffPOC(picA, picB) = POC(picA) + MaxPOC - POC(picB)
```

[0080] }The function POC(picX) is specified as follows: POC(picX) = pic_order_cnt of picX. POC_CYCLE_BD[i] is set to the poc_cycle of the i-th negative deltaPOC reference picture. The poc_cycle of the i-th negative deltaPOC reference picture is calculated based on a designated picture.

[0081] temporal_id_negative[i] specifies a temporal identifier and shall be represented by bits_for_temporal_id_in_buffer_descriptions bits, where bits_for_temporal_id_in_buffer_descriptions is the syntax element from the picture parameter set used by the current picture. temporal_id_negative[i] defines the value of the variable TemporalIDBD[i] as TemporalIDBD[i] = temporal_id_negative[i]. temporal_id_negative[i] shall be in the range of 0 to max_temporal_layers_minus1, inclusive. The value of TemporalIDBD[i] is restricted such that TemporalIDBD[i] must be equal to the temporal_id signaled in the NAL-header of the reference picture with pic_order_cnt equal to POCBD[i].

[0082] number_of_positive_pictures specifies the number of entries in the list DeltaPOCBD_pps[i] with a positive value. The value of number_of_positive_pictures shall be in the range of 0 to max_num_ref_frames-Offset, inclusive. When the number_of_positive_pictures syntax element is not present, the value of number_of_positive_pictures shall be inferred to be equal to 0.

[0083] delta_poc_minus_one[i] specifies an absolute distance in POC value. delta_poc_minus_one[i] defines the value of the variable POCBD[i+Offset] as POCBD[i+Offset] = (pic_order_cnt + (delta_poc_minus_one[i] + 1)) % MaxPOC if i equals 0 and as POCBD[i+Offset] = (POCBD[i-1+Offset] + (delta_poc_minus_one[i] + 1)) % MaxPOC if i > 0. Here, pic_order_cnt is the POC of the current picture as signaled in the slice header. The value of delta_poc_minus_one[i] shall be in the range of 0 to MaxPOC-1, inclusive. The value of POCBD[i+Offset] shall be such that the value of DiffPOC(currPic, refPic) is in the range of 1 to MaxPOC/2-1, where refPic is the reference picture with pic_order_cnt equal to POCBD[i+Offset]. POC_CYCLE_BD[i+Offset] is set to the poc_cycle of the i-th positive deltaPOC reference picture. The poc_cycle of the i-th positive deltaPOC reference picture is calculated based on a designated picture.

[0084] temporal_id_positive[i] specifies a temporal identifier and shall be represented by bits_for_temporal_id_in_buffer_descriptions bits, where bits_for_temporal_id_in_buffer_descriptions is the syntax element from the picture parameter set used by the current picture. temporal_id_positive[i] defines the value of the variable TemporalIDBD[i+Offset] as TemporalIDBD[i+Offset] = temporal_id_positive[i]. temporal_id_positive[i] shall be in the range of 0 to max_temporal_layers_minus1, inclusive. The value of TemporalIDBD[i] is restricted such that TemporalIDBD[i] must be equal to the temporal_id signaled in the NAL-header of the reference picture with pic_order_cnt equal to POCBD[i].

[0085] A bd_poc_cycle_update_flag equal to 1 specifies that the POC_CYCLE_BD_pps[bd_idx][j] of the referenced buffer description should be overridden for the current picture. In some configurations, future frames may also override poc_cycle information. If bd_poc_cycle_update_flag is 0 then the original POC_CYCLE_BD_pps[bd_idx][j] of the referenced buffer description are to be used. poc_cycle_pps_override[j] specifies the values to be used to override the values within the POC_CYCLE_BD_pps[bd_idx][j] for the current picture only. In an alternative configuration, poc_cycle_pps_override[j] specifies an offset. For the current picture only, (POC_CYCLE_BD_pps[bd_idx][j] + poc_cycle_pps_override[j]) may be used instead of POC_CYCLE_BD_pps[bd_idx][j].

[0086] poc_pps_override[j] specifies the values to be used to override the values within the POCBD_pps[bd_idx][j] for the current picture only. In an alternative configuration, poc_pps_override[j] specifies an offset. For the current picture only, (POCBD_pps[bd_idx][j] + poc_pps_override[j]) may be used instead of POCBD-pps[bd_idx][j]. temporal_id_pps_override[j] specifies the values to be used to override the values within the TemporalIDBD_pps[bd_idx][j] for the current picture only. In an alternative configuration, temporal_id_pps_override[j] specifies an offset. For the current picture only, (TemporalIDBD_pps[bd_idx][j] + temporal_id_pps_override[j]) may be used instead of TemporalIDBD_pps [bd_idx][j].

[0087] number_of_longterm_pictures specifies the number of long term picture entries in the list POCBD and POC_CYCLE_BD. The value of number_of_longterm_pictures shall be in the range of 0 to max_num_ref_frames, inclusive. max_num_ref_frames specifies the maximum number of short term and long term reference frames. poc[j] specifies a POC value to be added to the list POCBD as POCBD[bd idx][j] = poc[j]. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order. poc[j] shall be in the range of 0 to MaxPOC-1.

[0088] poc_cycle[j] (e.g., the cycle parameter) specifies poc_cycle value and defines the value to be added to the list POC_CYCLE_BD. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order. poc_cycle[j] may be less than or equal to zero or may occupy a different numerical range.

[0089] temporal_id_poc[j] specifies a temporal identifier and shall be represented by bits_for_temporal_id_in_buffer_descriptions bits. temporal_id_poc[j] defines the value to be added to the list TemporalIDBD. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order. temporal_id_poc[i][j] shall be in the range of 0 to max_temporal_layers_minus1, inclusive. max_temporal_layers_minus1+1 specifies the maximum number of temporal layers present in a sequence.

[0090] In some configurations, number_of_longterm_pictures_pps[bd-idx] may be transmitted before the "for" loop illustrated in Listing (3), thereby avoiding a dependency on a slice header with PPS. Alternatively, bd_poc_cycle_update_flag may be replaced with another parameter, num_longterm_poccycle_override_count. For example, relevant code in Listing (3) above may be replaced with "If num_longterm_poccycle_override_count> 0 then For(j = 0; j < num_longterm_poccycle_override_count ; j++) {....}."

[0091] A bd_reference_flag equal to 1 specifies that the buffer description of the current picture shall be created using syntax elements not transmitted in the slice header. In one configuration, the buffer description for the current picture is created using syntax elements in the picture parameter set. A bd_reference_flag equal to 0 specifies that the buffer description of the current picture shall be created by a combination of buffer descriptions transmitted in the slice header and buffer descriptions not transmitted in the slice header. In one configuration, the buffer descriptions not transmitted in the slice header are transmitted in the picture parameter set.

[0092] bd_idx identifies a buffer description in a set of available buffer descriptions. In one configuration, bd_idx specifies an index in the lists POCBD_pps, DeltaPOCBD_pps, POC_CYCLE_BD_pps, and TemporalIDBD_pps that shall be used for the creation of the buffer description of the current picture.

[0093] The values corresponding to deltaPOC referenced pictures in (DeltaPOCBD_pps[bd_idx], TemporalIDBD_pps[bd_idx]) are converted to picture order count (e.g., POC), cycle parameter (e.g., poc_cycle) and temporal identifier and added to the picture buffer description lists POCBD, POC_CYCLE_BD and TemporalIDBD. The values in the lists (POCBD_pps[bd_idx], POC_CYCLE_BD_pps[bd_idx], TemporalIDBD_pps[bd_idx]) corresponding to reference pictures are added to the picture buffer description lists POCBD, POC_CYCLE_BD and TemporalIDBD. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order.

[0094] A combine_with_reference_flag equal to 1 specifies that syntax elements from the active PPS are used to assign values in lists POCBD, POC_CYCLE_BD and TemporalIDBD in combination with explicit signaling. When the combine_with_reference_flag syntax element is not present, the value of combine_with_reference_flag shall be inferred to be equal to 0.

[0095] bd_combination_idx specifies an index in the lists DeltaPOCBD_pps and TemporalIDBD_pps that shall be used for creation of the buffer description of the current picture in combination with explicit signaling. bd_combination_idx shall be represented by ceil(log2(number_of_bds)) bits. The value of bd_combination_idx shall be in the range of 0 to number_of_bds-1, where number_of_bds is the syntax element from the picture parameter set used by the current picture. The current pictures in POCBD, POC_CYCLE_BD and TemporalIDBD lists are then added values from the lists DeltaPOCBD_pps[bd_idx], POCBD_pps[bd_idx], POC_CYCLE_BD_pps[bd_idx] and TemporalIDBD_pps[bd_idx] using a pre-defined mechanism. In one configuration of the above, the addition operation is an appending operation. In another configuration, the appending (or addition) operation is an operation that replaces items in the list in a predefined order.

[0096] Some examples of ways in which the systems and methods described herein may be applied are given hereafter. Assume that a picture with POC = 0 on is a long-term (reference) picture used by a picture with POC = *MaxPOC* -1 and a picture with POC = 0 from a subsequent picture set. The long-term (reference) picture may be indicated in different ways.

[0097] In a first way, there are two buffer descriptions in the PPS, including buffer description A: {POC=0,poc_cycle=0,temporalID} and buffer description B: {POC=0,poc_cycle=-1,temporalID}. The picture with POC = *MaxPOC* -1 will point to buffer description A. The picture with POC = 0 from the subsequent picture set will refer to buffer description B.

[0098] In a second alternative way, the picture with POC = *MaxPOC* -1 will point to buffer description A. The picture with POC = 0 from the subsequent picture set will refer to buffer description A. The buffer description A corresponds to {POC=0, poc_cycle=0, temporalID}. For referencing the correct picture, the poc_cycle should be set to -1, since the referenced picture belongs to the previous [0, ···, *MaxPOC* -1] picture set. Thus, the poc_cycle value for buffer description A that is currently set to 0 may be overridden (e.g., set to a different value) for the current slice only, by transmitting a "-1" in the slice header. In some configurations, the first slice in a picture may be used to override the poc_cycle value of a buffer description for the current picture only.

[0099] In some configurations, the list of reference pictures signaled at a high level may be modified at a finer level. For example, a list of reference pictures may be signaled in the buffer description transmitted in the PPS (e.g., signaled at high level). However, the buffer description transmitted in the PPS may be modified (e.g., modified at a finer level). For example, the buffer description may be modified by deleting existing entries. Additionally or alternatively, the buffer description information from the PPS may be modified to add new entries or replace an entry. The buffer description information from the PPS may also be modified to add new entries and replace current entries using a predefined mechanism, such as if no empty slots are available. A benefit of allowing modifications is that a higher level of control may be obtained. This may be instead of using the buffer descriptions from the PPS without modification or in addition to or alternatively by replacing only the first 'n' entries.

[0100] In some configurations, additional syntax may be defined to replace entries in the slot list. In this case, the index in the list that is replaced and the entry that may replace the current entry at that index will be specified.

[0101] Additionally or alternatively, additional syntax may be defined to add entries in the list with replacement if the slot list has no vacancies. In this case, the index in the list that should be replaced may be signaled explicitly (from the encoder 108 or overhead signaling module 112, for example) or inferred implicitly (by the decoder 102, for example) using some information transmitted earlier in the bitstream.

[0102] Listing (4) illustrates another example of syntax modification for the PPS from AHG21. In particular, Listing (4) illustrates another example of buffer description syntax used in slice headers as outlined in AHG21. However, modifications to the syntax given in AHG21 in accordance with the systems and methods disclosed herein are denoted in bold text in Listing (4).

```
/*  Buffer description
*/
buffer_description( ) {
    bd_reference_flag
    if(bd_reference_flag = = 1) {
        bd_idx
        bd_poc_cycle_update_flag
        if (bd_poc_cycle_update_flag == 1)
            for( j = 0; j < number_of_longterm_pictures_pps[bd_idx]; j++ ) {
                poc_pps_override[j]
                poc_cycle_pps_override[j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_pps_override[j]
        }
    bd_poc_cycle_delete_count
    if (bd_poc_cycle_delete_count >0 )
        for( j = 0; j < bd_poc_cycle_delete_count; j++ ) {
            bd_pps_delete_idx[j]
        }
    bd_poc_cycle_append_count
    if (bd_poc_cycle_append_count >0 )
        for( j = 0; j < bd_poc_cycle_append_count; j++ ) {
            poc_pps_append[j]
            poc_cycle_pps_append[j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_pps_append[j]
        }
} else {
    number_of_negative_pictures
    for( i = 0; i < number_of_negative_pictures; i++ ) {
    ...
    }
    if( positive_pictures_in_buffer_descriptions_flag ){
    ...
    }
    number_of_longterm_pictures
    for( j = 0; j < number_of_longterm_pictures; j++ ) {
        poc[j]
        poc_cycle[j]
        if( bits_for_temporal_id_in_buffer_descriptions > 0 )
            temporal_id_poc[j]
    }
```

```
        if( number_of_negative_pictures + number_of_positive_pictures <
          max_num_ref_frames ) {
          combine_with_reference_flag
          if( combine_with_reference_flag )
            bd_combination_idx
        }
      }
    }
```

Listing (4)

[0103] Listing (4) contains many elements similar to that of Listing (3). Listing (4) also contains syntax modifications to the slice header to allow for deletions and/or additions of reference pictures.

[0104] Examples of descriptions of the parameters in Listing (4) are given as follows. A bd_poc_cycle_delete_count specifies the number of entries to be deleted from the lists: POCBD_pps[bd_idx], temporalIDBD_pps[bd_idx] and POC_CYCLE_BD_pps[bd_idx] for the current picture (only, for example). In some configurations, bd_poc_cycle_delete_count may be greater than or equal to 0.

[0105] bd_pps_delete_idx[j] specifies that, for the current picture, the POC_CYCLE_BD_pps[bd_idx][bd_pps_delete_idx[j]] entry, temporalIDBD_pps [bd_idx][bd_pps_delete_idx[j]] entry and the POCBD_pps [bd_idx][bd_pps_delete_idx[j]] entry are deleted in the lists. In some configurations, once a picture is deleted, it may no longer be available for reference by subsequent decoded pictures. In other configurations, once all the deletions are completed, the remaining entries in the list POCBD_pps[bd_idx], temporalIDBD_pps[bd_idx] and POC_CYCLE_BD_pps[bd_idx] may be moved towards the initial index (e.g., zero-th index) to occupy the vacated slots. This process may be carried out so no empty slots remain between any two occupied slots. Here, a slot corresponds to an entry in the buffer description. For example, if the list is [Slot0=A] [Stot1=empty] [Slot2=B] [Slot3=C] [Slot4=empty] [Slot 5=D], then after the deletions the entries are moved towards Slot0 until no empty slots remain between any two occupied slots. In this example, the resulting list is [Slot0=A] [Slot1=B] [Slot2=C] [Slot3=D] [Slot4=empty] [Slot 5=empty].

[0106] bd_poc_cycle_append_count specifies the number of entries to be appended to lists: POCBD_pps[bd_idx], temporalIDBD_pps[bd_idx] and POC_CYCLE_BD_pps[bd_idx] for the current picture. In some configurations, bd_poc_cycle_append_count may be greater than or equal to 0.

[0107] poc_pps_append[j] specifies the POC value to be appended to the list POCBD_pps[bd_idx] for the current picture (only, for example). In some configurations, if any empty slots exist in the list, additions may be performed starting with the empty slot closest to the zero-index and then in increasing order of slot index.

[0108] poc_cycle_pps_append[j] specifies the poc_cycle value to be appended to the list POC_CYCLE_BD_pps[bd_idx] for the current picture (only, for example). In some configurations, if any empty slots exist in the list, then the empty slot closest to the zero-index may be filled first. Additionally, slots may be filled in increasing order of slot index. In some configurations, these operations of overriding entries, deleting entries and appending entries to POCBD_pps[bd_idx] and POC_CYCLE_BD_pps[bd_idx] are carried out in a pre-defined order.

[0109] temporal_id_pps_append[j] specifies the temporal identifier value to be appended to the list TemporalIDBD_pps[bd_idx] for the current picture (only, for example). In some configurations, if any empty slots exist in the list, additions may be performed starting with the empty slot closest to the zero-index and then in increasing order of slot index.

[0110] Some examples of configurations of the systems and methods disclosed herein are given hereafter. In one configuration, the 3-tuple (POC, poc_cycle, temporalID) may be replaced with the 2-tuple (LTSlotIdx, temporalID). LT-SlotIndex may be a slot index that points to a location in the long-term DPB. One possible benefit of this approach is to reduce bitrate overhead.

[0111] In another configuration, the 3-tuple (POC, poc_cycle, temporalID) may be replaced with (f(POC, poc_cycle), temporalID), where f(POC, poc_cycle) is a function (e.g., look-up table) that maps the two-tuple (POC, poc_cycle) to an index. In yet another configuration, instead of (POC, poc_cycle), an absolute POC value may be used. In one example, this absolute POC value will be different than the longterm_poc[i] field in a buffer description in JCTVC-F493 that specified an absolute POC. In particular, the JCTVC-F493 longterm_poc[i] field could only signal values in the range 0 and MaxPOC. In accordance with the systems and methods disclosed herein, however, the absolute POC value may be signaled (and possibly derived) using a most significant bit (MSB) and least significant bit (LSB) scheme of AVC, thus providing larger bit resolution. Alternatively, the absolute POC value may be signaled using a number space that is larger than MaxPOC.

[0112] In another configuration, the absolute POC in accordance with systems and methods disclosed herein may be used for referencing all reference pictures (e.g., short term pictures and long term pictures). In another configuration, the absolute POC in accordance with the systems and methods disclosed herein may be used instead of deltaPOC signaling. In some configurations, POC and poc_cycle may be replaced by an absolute POC value as shown in example hereafter (and may not be llimited to the examples below):

```
/* Picture parameter set raw byte sequence payload (RBSP) syntax
*/
pic_parameter_set_rbsp( ) {
   ...
            number_of_longterm_pictures_pps[i]
            for( j = 0; j < number_of_longterm_pictures_pps[i]; j++ ) {
               absolute_poc_pps[i][j]
               if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                  temporal_id_poc_pps[i][j]
            }
         }
   }
   ...
}
```

[0113] In this example, **absolute_poc_pps[i][j]** is the absolute picture order count for the j-th long term picture entry in the i-th buffer description within the PPS.

```
/* Buffer description
*/
buffer_description( ) {
   bd_reference_flag
   if(bd_reference_flag = = 1) {
      bd_idx
...
   } else {
...
      number_of_longterm_pictures
      for( j = 0; j < number_of_longterm_pictures; j++ ) {
            absolute_poc[j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_poc[j]
        }
...
    }
}
```

[0114] In this example, **absolute_poc[j]** is the absolute picture order count for the j-th long term picture entry in buffer description.

[0115] In some configurations, some or all information typically contained in the PPS and/or in buffer descriptions may be additionally or alternatively carried in an Adaptive Slice Parameter Set or Adaptation Parameter Set (APS). This information includes one or more of: number_of_longterm_pictures, poc[j], poc_cycle[j], temporal_id_poc[j], number_of_longterm_pictures_pps[i], poc_pps[i][j], poc_cycle_pps[i][j] and temporal_id_poc_pps[i][j]. For example, The Adaptive Slice Parameter Set or Adaptation Parameter Set (APS) may include one or more of a number of reference pictures (e.g., number_of_longterm_pictures), a picture order count (e.g., poc[j]), a picture order count cycle parameter (e.g., poc_cycle[j]), a temporal identifier picture order count parameter (e.g., temporal_id_poc[j]), a picture parameter set number of reference pictures (e.g., number_of_longterm_pictures_pps[i]), a picture parameter set picture order count (e.g., poc_pps[i][j]), a picture parameter set picture order count cycle parameter (e.g., poc_cycle_pps[i][j]) and a picture parameter set temporal identifier picture order count parameter (e.g., temporal_id_poc_pps [i][j]).

[0116] In some configurations, the information poc_cycle[j] may only be signaled (e.g., from an encoder 108 to the decoder 102, 202) if it is different than 0. In this case, an alternate syntax may be defined.

[0117] In an yet another configuration, some or all information typically contained in the PPS and/or in the buffer descriptions may additionally or alternatively be carried in a slice header separately from the buffer description information. For example, the slice header may carry (separately from the buffer description container) one or more of a number of reference pictures (e.g., number_of_longterm_pictures), a picture order count (e.g., poc[j]), a picture order count cycle parameter (e.g., poc_cycle[j]), a temporal identifier picture order count parameter (e.g., temporal_id_poc[j]), a picture parameter set number of reference pictures (e.g., number_of_longterm_pictures_pps[i]), a picture parameter set picture order count (e.g., poc_pps[i][j]), a picture parameter set picture order count cycle parameter (e.g., poc_cycle_pps[i][j]) and a picture parameter set temporal identifier picture order count parameter (e.g., temporal_id_poc_pps[i][j]).

[0118] In an alternative configuration, a long term (reference) picture may be signaled by indexing it as x.y, where x = poc[j] or poc_pps[i][j] and y is a new information subindex that defines an additional namespace/numberspace for subindexing long term (reference) pictures. In this case, the x and y entries may be sent in PPS and/or buffer descriptions (in a slice header) for each long term (reference) picture.

[0119] In some configurations, all (reference) pictures (e.g., long-term and short-term) are referenced using either delta referencing (using deltaPOC and temporalID, for example) or absolute referencing (using POC, poc_cycle and temporalID, for example). For example, the entire decoded picture buffer (DPB) may contain a set of received pictures. A subset of these received pictures may use delta referencing and the remaining received pictures may use absolute referencing. It should be noted that prior approaches do not specify the same absolute referencing as given in accordance with the systems and methods disclosed herein (using POC and poc_cycle, for example). It should be noted that one or more of the configurations of buffer descriptions and syntaxes described may be implemented in combination with one or more of the methods and/or approaches described herein.

[0120] Figure 3 is a flow diagram illustrating one configuration of a method 300 for tracking a reference picture with reduced overhead referencing based on a designated picture. An electronic device 204 (e.g., decoder 202) may receive 302 a bitstream. For example, the decoder 202 may receive 302 a bitstream 214 that includes an encoded reference picture (and other encoded pictures, for instance). In some configurations, the bitstream 214 may also include overhead information (e.g., PPS, buffer description information, parameters, wrap indicators, reference picture designation or identifier, etc.).

[0121] The electronic device 204 may decode 304 a portion of the bitstream 214 to produce a decoded reference picture. For example, the decoder 202 may decode 304 a portion of the bitstream 214 to produce a decoded reference picture that is stored in frame memory 264. It should be noted that one or more portions of the bitstream 214 may be decoded 304 to produce one or more decoded reference pictures.

[0122] The electronic device 204 may track 306 the decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing.

[0123] For example, the electronic device 204 may associate a cycle parameter with the decoded reference picture and modify (e.g., decrement or increment) the cycle parameter if a wrap indicator is received or if a transition between picture sets is determined. Other approaches may be used for tracking 306 the decoded reference picture. Greater detail is given below. It should be noted that the DPB may include one or more decoded reference pictures.

[0124] The electronic device 204 may decode 308 the picture based on one or more decoded reference pictures. For example, a portion of the bitstream 214 (other than the portion decoded 304 to produce the decoded reference picture) may be decoded 308 based on the reference picture. For instance, the decoded reference picture (that has been tracked in the DPB) may be provided to a motion compensation module 260 in order to generate an inter-frame prediction signal 268 based on an inter-frame prediction mechanism. The inter-frame prediction signal 268 may then be used to decode 308 the picture. In some configurations or instances, one or more decoded reference pictures may be tracked 306 and used to decode 308 the picture.

[0125] Figure 4 is a flow diagram illustrating a more specific configuration of a method 400 for tracking a reference picture with reduced overhead referencing based on a designated picture. This method 400 may be one approach for tracking which picture is being referenced when POCs are reused. An electronic device 204 (e.g., decoder 202) may receive 402 a bitstream 214. For example, the decoder 202 may receive 402 a bitstream 214 that includes an encoded reference picture (and other encoded pictures, for instance). In some configurations, the bitstream 214 may include overhead information (e.g., PPS, buffer description information, parameters, wrap indicators, reference picture designation or identifier, etc.).

[0126] The electronic device 204 may decode 404 a portion of the bitstream 214 to produce a decoded reference picture. For example, the decoder 202 may decode 404 a portion of the bitstream 214 to produce a decoded reference picture that is stored in frame memory 264. It should be noted that one or more portions of the bitstream 214 may be decoded 404 to produce one or more decoded reference pictures.

[0127] The electronic device 204 may associate 406 a cycle parameter with a decoded picture set that includes the decoded reference picture. For example, the electronic device 204 may associate 406 a cycle parameter "poc_cycle" with a decoded picture set that includes the decoded reference picture.

[0128] The cycle parameter "poc_cycle" may be defined as follows. When a fixed number of bits are used to represent the POC of a picture in a range [0, ···, *MaxPOC*-1], *MaxPOC* unique integer values exist. If the number of pictures being encoded exceeds MaxPOC , a picture numbering mechanism must reuse already assigned POC values. The POC numbering then progresses as follows in one example: ···, $[0, \cdots, MaxPOC\text{-}1]_{n-2}$, $[0,\cdots, MaxPOC\text{-}1]_{n-1}$, $[0,\cdots, MaxPOC\text{-}1]_{n}$, $[0,\cdots, MaxPOC\text{-}1]_{n+1}$,···. The subscript in this example denotes the number of times the set [0,···, *MaxPOC*-1] has been repeated. This subscript or the number of times the set [0,···, *MaxPOC*-1] has been repeated may be denoted as *MaxPOCSetIndex*. For example, a picture with POC = 0 and *MaxPOCSetIndex* = *n* represents the $(n^{*}\ MaxPOC + 1)^{th}$ picture of the sequence (with an assumption that picture set numbering starts with 1, for instance).

[0129] Additional detail regarding the cycle parameter "poc_cycle" is given in connection with Figure 5 below.

[0130] The electronic device 204 may determine 408 whether a wrap indicator is received. For example, each time an encoder 108 or transmitting electronic device A 104a reaches a predetermined maximum number of pictures in a set of pictures, the encoder 108 or transmitting electronic device A 104a may send a wrap indicator that is received by the decoder 102 or receiving electronic device B 104b to indicate that another set of pictures is being sent (e.g., a POC is resetting or starting another cycle). Greater detail is given in connection with Figure 12 below.

[0131] If the electronic device 204 determines 408 that a wrap indicator was received, the electronic device 204 may modify 410 (e.g., decrement) the cycle parameter. For example, the electronic device 204 decrements cycle parameters for each picture or each set of pictures in the DPB. In another example, the electronic device 204 may increment the cycle parameter.

[0132] The electronic device 204 may decode 412 a picture based on the decoded reference picture. For example, a portion of the bitstream 214 (other than the portion decoded 404 to produce the decoded reference picture) may be decoded 412 based on the reference picture. For instance, the decoded reference picture (that has been tracked in the DPB) may be provided to a motion compensation module 260 in order to generate an inter-frame prediction signal 268 based on an inter-frame prediction mechanism. The inter-frame prediction signal 268 may then be used to decode 412 the picture. In some configurations or instances, one or more decoded reference pictures may be used to decode 412 the picture.

[0133] Figure 5 is a diagram illustrating one example of multiple picture sets referenced by cycle parameters. More specifically, Figure 5 illustrates an example of tracking a reference picture with reduced overhead referencing based on a designated picture using a cycle parameter. In particular, Figure 5 illustrates a cycle parameter (e.g., poc_cycle = -1) associated with picture set A 507a, a cycle parameter (e.g., poc_cycle = 0) associated with picture set B 507b and a cycle parameter (e.g., poc_cycle = +1) associated with picture set C 507c. However, it should be noted that picture set A 507a may or may not be the first picture set in a sequence of frames. For example, one or more picture sets may precede picture set A 507a. Furthermore, it should be noted that picture set C 507c may or may not be the last picture set in a sequence of frames. For example, one or more picture sets may follow picture set C 507c.

[0134] Each picture set 507a-c may include one or more pictures 501a-n, 503a-n, 505a-n. In this example, each picture set 507a-c includes *MaxPOC* pictures 501, 503, 505. In particular, each picture 501, 503, 505 may have a corresponding picture order count (POC), denoted as [0, 1, 2,···, *MaxPOC*-1] in Figure 5.

[0135] In one example, the poc_cycle of the picture set that contains the current decoded picture may be set to 0 for computing the poc_cycle of other pictures. In some cases, pictures may be decoded out of order. For example, a decoder may see 503b, then 505a and then 503c. In this example, assume that a picture being currently decoded is a picture 503b in picture set B 507b with POC = 1. The poc_cycle of another picture, such as a reference picture, may then be calculated based on the poc_cycle of the current decoded picture.

[0136] Figure 6 is a diagram illustrating another example of multiple picture sets. Figure 6 contains picture sets 607a-c similar to the picture sets 507a-c described above in connection with Figure 5. One or more pictures 601a-c, 603a-n, 605a-n may be included in each picture set 607a-c. A picture set 607a-c may include *MaxPOC* pictures. Further, each picture 601a-c, 603a-603n, 605a-605n may have a corresponding picture order count (POC), belonging to the set [0,1,2, ···, *MaxPOC* -1] as denoted in Figure 6.

[0137] A picture set 607 may include one or more pictures. For example, pictures 603a-n in picture set B 607b may be grouped as a picture set. Likewise, pictures 605a-n in picture set C 607c may be grouped together into a picture set. A picture set may include pictures that are adjacent to each other when encoded or may consist of random pictures grouped together. In another example, a picture set may make up a poc_cycle consisting of [0,1,2,···, *MaxPOC*-1] pictures.

[0138] In some configurations, a picture set 607a may include an instantaneous decoding refresh (IDR) picture 603a (e.g., IDR access unit). The IDR picture 603a may be signaled by the encoder in the bitstream. Additionally or alternatively, the IDR may be identified based on picture type. In some configurations, the IDR picture 603a may indicate to the decoder 202 that no subsequent pictures in the bitstream 214 will require reference to prior pictures in the bitstream 214.

[0139] When an IDR picture 603a is decoded by a decoder 202, the POC may be reset back to a predetermined value. For example, after decoding picture set A 607a, an IDR picture 603a is received or signaled. At this point, the POC may be reset back to 0 and a new picture set, picture set B 607b, begins. In other words, the IDR picture 603a may specify the reference picture relative to a designated picture. One benefit of sending an IDR picture 603a may be to introduce a new reference picture. This benefit will be described in greater detail below in connection with Figure 7.

[0140] Figure 7 is a diagram of a more specific example of multiple picture sets referenced by cycle parameters. Figure 7 illustrates picture sets 707a-d and pictures 701, 703, 705 along with their associated POC and poc_cycle similar to components 507a-c, 501, 503, 505, described above in connection with Figure 5. Additionally, Figure 7 shows an addition picture set 707d along with associated pictures 731a-n.

[0141] Figure 7 also illustrates a designated picture 703a. In one configuration, the cycle parameter (e.g., poc_cycle) for the designated picture may be set to 0. For example, when the first picture in a picture set (e.g., POC = 0) is a designated picture (e.g., IDR picture), then the poc_cycle for the designated picture may be equal to 0. The cycle parameter (e.g., poc_cycle) of later picture sets may be calculated with respect to the picture set with the cycle parameter set to 0 (e.g., poc_cycle = 0).

[0142] One example of a designated picture is an IDR picture 703a (e.g., IDR access unit) similar to the IDR picture 603a described above in connection with Figure 6. For convenience, the designated picture 703a will be described as an IDR picture. However, it should be noted that the designated picture 703a may be any picture that is indicated by the encoder with signaling in the bitstream or that may be implicitly determined by the decoder.

[0143] In some configurations, an encoder may send a signal in the bitstream that indicates the IDR picture 703a. The value of the POC counter may be determined based on this signal. For example, the signal may indicate (to the decoder) to set the POC to 0 for the IDR picture (if the POC is not 0 for the IDR picture, for example). For instance, the IDR picture 703a may cause a decoder to reset the POC counter to the initial value, 0 for instance. The poc_cycle may also be based on the IDR picture 703a. Similar to the POC, the signal may indicate that the poc_cycle should be reset to the initial value. In another example, the signal may indicate that no modification to the POC or the poc_cycle need occur (if they are at the initial value for the IDR picture 703a).

[0144] The IDR picture 703a may be an indication to set or reset the POC and poc_cycle (and/or other cycle parameters, for example), which may be used to track a reference picture relative to the designated picture 703a. For example, the decoder 202 may decode pictures 701a-c in set A 707a. Picture set A 707a may have the cycle parameter poc_cycle = m associated with it. An IDR picture 703a may then be signaled from the bitstream 214. The IDR picture 703a may indicate a new cycle parameter (e.g., poc_cycle = 0) and may indicate a POC = 0. Picture sets 707b-d may then be decoded and their associated cycle parameter may increment from the initial value (e.g., poc_cycle = 0, poc_cycle = +1, poc_cycle = +2, respectively). In this example, later picture sets 707b-d are calculated with respect to the newly set cycle parameter and POC.

[0145] As described previously, an IDR picture 703a may indicate to the decoder 202 that no subsequent pictures in the bitstream will require reference to pictures prior pictures in the bitstream 214. However, an IDR picture 703a may just signal a reset of the POC or of the cycle parameters. For example, picture set A may have a cycle parameter of poc_cycle = m where m = +5. When the IDR picture 703a is received, m may be changed to be m = -1. Thus, the IDR picture 703a may make the poc_cycle count of the reference picture relative to a designated picture. In other words, the IDR picture 703a may signal the poc_cycle count to calculate the reference picture relative to a designated picture in the sequence, not just the current picture.

[0146] Figure 8 is a flow diagram illustrating one configuration of a method 800 for tracking a reference picture with reduced overhead referencing based on a designated picture. An electronic device 204 (e.g., decoder 202) may receive 802 a bitstream and decode 804 a portion of the bitstream 214 to produce a decoded reference picture. This may occur in a similar manner as described previously in connection to Figure 3.

[0147] The electronic device 204 may track the decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing based on a designated picture. This may be accomplished as follows. The electronic device 204 (e.g., decoder 202) may determine 806 a designated picture. In one configuration, the electronic device 204 may make this determination 806 based on explicit signaling received in the bitstream 214. For instance, the bitstream 214 may include an indicator that specifies the designated picture. In one example, the designated picture may be an instantaneous decoding refresh (IDR) picture that is indicated by signaling (e.g., in a picture parameter set (PPS), in a buffer description, etc.) received in the bitstream 214.

[0148] The electronic device 204 may determine 808 a picture order count (POC) based on the designated picture. For example, the electronic device 204 may set or reset the POC associated with the designated picture. For instance, the POC associated with the designated picture may be set to 0. Accordingly, the electronic device 204 may reset a POC sequence based on the designated picture. For example, the POCs for other pictures may be numbered based on the POC associated with the designated picture 603a as illustrated in Figure 6.

[0149] The electronic device 204 may determine 810 a cycle parameter (e.g., poc_cycle) based on the designated picture. For example, the cycle parameter associated with a picture set that includes the designated picture may be set or reset to 0. The cycle parameters for other picture sets may be determined (e.g., calculated) based on the picture set including the designated picture (e.g., IDR picture). One example of this is illustrated in and described in connection with Figure 7.

[0150] When the POCs and cycle parameters (e.g., poc_cycles) are determined based on the designated picture, the POC and cycle parameter of the decoded reference picture may be updated as necessary. This may enable tracking of the decoded reference picture in the decoded picture buffer (DPB). Thus, the decoded reference picture may be tracked based on the designated picture (e.g., IDR picture).

[0151] The electronic device 204 may obtain 812 a decoded reference picture from the DPB. For example, the electronic device 204 may retrieve the decoded reference picture from the DPB according to a POC and cycle parameter (e.g., poc_cycle) that are based on the designated picture.

[0152] The electronic device 204 may decode 814 a picture based on the decoded reference picture. This may occur as similarly described above in connection with Figure 3. It should be noted that a set of one or more decoded reference pictures may be tracked in accordance with the method 800.

[0153] Figure 9 is a flow diagram illustrating another configuration of a method 900 for tracking a reference picture with reduced overhead referencing based on a designated picture. As described preciously, an electronic device 204 may receive 902 a bitstream. The electronic device 204 may also decode 904 one or more portions of the bitstream to produce a reference picture collection. The decoded reference picture collection may include one or more decoded reference pictures.

[0154] The electronic device 204 may track 906 the decoded reference picture collection in a decoded picture buffer (DPB) with reduced overhead referencing based on a designated picture. This may be done as described above in connection with Figure 8. Two or more decoded reference pictures in the decoded reference picture collection may refer to the same or different time instances (e.g., similar time instances) as described above.

[0155] The electronic device 204 may optionally determine 908 whether the resolution of a decoded reference picture (for decoding the picture) in the decoded reference picture collection is different from a resolution of the current picture. If the resolution between the two pictures is different, then the electronic device 204 may process 910 the transform coefficients of the decoded reference picture based on a scaling parameter to decode the picture. A more detailed description regarding the scaling parameter and resolution switching is given above in connection with Table (2). It should be noted that the scaling parameter may be determined by receiving the scaling parameter explicitly or by determining the scaling parameter implicitly (using a look-up table, picture resolution characteristics, etc., for example).

[0156] Regardless of whether the resolutions are the same or different, the electronic device 204 may decode 912 a picture based on the decoded reference picture collection (e.g., a decoded reference picture in the decoded reference picture collection).

[0157] Figure 10 is a flow diagram illustrating another configuration of a method 1000 for tracking a reference picture with reduced overhead referencing based on a designated picture. An electronic device 204 may receive 1002 a bitstream and decode 1004 a portion of the bitstream to produce a decoded reference picture. The electronic device 204 may track 1006 the decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing based on a designated picture. This may be done as described above.

[0158] The electronic device 204 may also obtain 1008 a buffer description. Some examples of buffer descriptions are described in detail above. For instance, Table (1) and Table (2) above contain examples of various buffer descriptions. The buffer descriptions may be received by a decoder as part of the parameter picture set (PPS) from an encoder. An index to the appropriate buffer descriptions may also be received by a decoder as a part of a slice header. Once the buffer description is obtained 1008, the electronic device 204 may determine 1010 whether to modify the buffer description. If the buffer description requires no modification, then the electronic device 204 may decode 1014 a picture based on the decoded reference picture. However, if the buffer description requires modification, then the electronic device 204 may modify 1012 one or more entries. Examples of modifying 1012 may include deleting an entry, adding an entry or replacing an entry. A more detailed description of deleting, adding or replacing entries may be found above in connection with the description of Listing (4). Once a modification 1012 is complete, the electronic device 204 may decode 1014 a picture based on the decoded reference picture.

[0159] Figure 11 is a diagram illustrating one example of signaling a wrap indicator in accordance with the systems and methods disclosed herein. In this example, several pictures 1101a-n, 1103a are illustrated. The first picture 1101a with POC = 0 is a reference picture for the remainder of the pictures 1101b-n, 1103a illustrated in Figure 11. In particular, Figure 11 illustrates an association or correspondence 1109 between the reference picture 1101a and the other pictures 1101b-n, 1103a. For example, the picture 1101a with POC = 0 may be a long term reference picture 1101a to be kept in the DPB for decoding other pictures 1101b-n, 1103a.

[0160] As illustrated in Figure 11, POC numbers 0 through *MaxPOC* -1 and a reused 0 may respectively correspond to the pictures 1101a-n, 1103a. A first set of pictures 1101a-n may correspond to POC numbers 0 through *MaxPOC* -1. As described above, each set of pictures (with POC numbers 0 through *MaxPOC* -1) may correspond to a cycle parameter (e.g., poc_cycle).

[0161] In one configuration, a wrap indicator may be signaled 1133 at the first transition between one picture set and a subsequent later picture set. For example, the first time the POC numbering transitions from one [0,···, *MaxPOC* -1] set to the next, the wrap indicator may be signaled 1133. In some configurations, the wrap indicator signaled may be a protected message denoted "poc_wraparound." As used herein, "signaled" may mean communicated between an encoder and a decoder. In some configurations, "signaled" may also mean communicated between different electronic devices.

[0162] A protected message may be a message that must be received by the electronic device 204 in order to maintain a desired functionality such as detection of lost pictures. One mechanism to transmit a message as a protected message is to assign a higher priority to the protected message when compared to other information messages. An intelligent device (e.g., a network congestion control agent) may then examine this priority assignment and drop lower priority messages to meet constraints such as available network bandwidth.

[0163] In some configurations, the wrap indicator (e.g., poc_wraparound) message may be signaled in the Picture Parameter Set (PPS), Slice Header, Adaptive Parameter Set (APS) or any suitable location in the bitstream. Additionally or alternatively, the wrap indicator may be signaled out-of-band (e.g., separate from the picture bitstream). Each time the wrap indicator (e.g., poc_wraparound message) is received by the decoder 102, the cycle parameter (e.g., poc_cycle) for every picture (e.g., every picture set) in the DPB may be decremented (by 1, for example).

[0164] Figure 12 is a diagram illustrating one example of signaling a wrap indicator 1133 in accordance with the systems and methods disclosed herein. In this example, several pictures 1137a-n, 1139a-n, 1141a-n, 1143a-n, 1145a-n and picture sets 1107a-e are illustrated. In particular, Figure 12 illustrates a reference buffer status 1135 over time, along with corresponding current frames and the timing of the wrap indicator 1149.

[0165] In this example, assume that a picture B_0 1139a in picture set B 1107b with poc_cycle = 0 is the current picture or frame at a first time. When transition A 1147a occurs, a wrap indicator is signaled 1133a. At a second time, a picture C_0 1141a in picture set C 1107c is the current picture or frame. Assume that picture C_0 1141a is a reference picture for all subsequent pictures (e.g., in picture set D 1107d and picture set E 1107e). For instance, picture C_0 1141a may be a long term reference picture to be kept in the DPB for decoding subsequent pictures 1141b-n, 1143a-n, 1145a-n. When transition B 1147b occurs, another wrap indicator is signaled 1133b. As illustrated, the poc_cycle parameters are updated upon the signaling 1133 of the wrap indicators. This procedure may be used to track a reference picture in accordance with the systems and methods disclosed herein. Furthermore, it should be noted that the POC and poc_cycle may be reset or restarted based on a decoded reference picture as described herein.

[0166] As illustrated in Figure 12, POC numbers 0 through *MaxPOC*-1 repeatedly correspond to the pictures 1137a-n, 1139a-n, 1141a-n, 1143a-n, 1145a-n. As described above, each picture set 1107a-e (with POC numbers 0 through *MaxPOC* -1) may correspond to a cycle parameter (e.g., poc_cycle).

[0167] In one configuration, a wrap indicator may be signaled 1133a at the first transition 1147a between a current picture set (e.g., picture set B 1107b) and a subsequent later picture set (e.g., picture set C 1107c). For example, the first time the POC numbering transitions from one [0,···, *MaxPOC* -1] set to the next, the wrap indicator may be signaled

1133. In some configurations, the wrap indicator may be signaled 1133 when POC numbering transitions from one [0, ···, *MaxPOC* -1] set to another. Alternatively, the wrap indicator may be signaled 1133 the first time the poc_cycle numbering transitions for the next [0,···, *MaxPOC*-1] picture set. In some configurations, the wrap indicator may be signaled 1133 when poc_cycle numbering transitions for the next [0,···, *MaxPOC*-1] picture set. In some configurations, the wrap indicator signaled may be a protected message denoted "poc_wraparound." As used herein, "signaled" may mean communicated between an encoder and a decoder. In some configurations, "signaled" may also mean communicated between different electronic devices.

[0168] A protected message may be a message that must be received by the electronic device 204 in order to maintain a desired functionality such as detection of lost pictures. One mechanism to transmit a message as a protected message is to assign a higher priority to the protected message when compared to other information messages. An intelligent device (e.g., a network congestion control agent) may then examine this priority assignment and drop lower priority messages to meet constraints such as available network bandwidth.

[0169] In some configurations, the wrap indicator (e.g., poc_wraparound) message may be signaled in the Picture Parameter Set (PPS), Slice Header, Adaptation Parameter Set (APS) or any suitable location in the bitstream. Additionally or alternatively, the wrap indicator may be signaled out-of-band (e.g., separate from the picture bitstream). Each time the wrap indicator (e.g., poc_wraparound message) is received by the decoder 102, the cycle parameter (e.g., poc_cycle) for every picture (e.g., every picture set) in the DPB may be decremented (by 1, for example).

[0170] Listing (5) below illustrates an example of a bitstream syntax modification required to signal the wrap indicator in the picture parameter set:

```
/*  Picture parameter set RBSP syntax
*/
pic_parameter_set_rbsp( ) {
    seq_parameter_set_id
    pic_parameter_set_id
    entropy_coding_mode_flag
    wrap_indicator_flag
    bits_for_temporal_id_in_buffer_descriptions
    positive_pictures_in_buffer_descriptions_flag
    number_of_bds
    if( number_of_bds > 0 ) {
        for(i = 0; i < number_of_bds; i++){
            number_of_negative_pictures_pps[i]
            for(j = 0; j < number_of_negative_pictures_pps[i]; j++ ) {
                negative_delta_poc_minus_one_pps[i][j]
                if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                    temporal_id_negative_pps[i][j]
            }
            if(positive_pictures_in_buffer_descriptions_flag ){
                number_of_positive_pictures_pps[i]
                for( j = 0; j < number_of_positive_pictures_pps[i]; j++ ) {
                    delta_poc_minus_one_pps[i][j]
                    if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                        temporal_id_positive_pps[i][j]
                }
            }
            number_of_longterm_pictures_pps[i]
            for( j = 0; j < number_of_longterm_pictures_pps[i]; j++ ) {
                poc_pps[i][j]
                poc_cycle_pps[i][j]
                if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                    temporal_id_poc_pps[i][j]
            }
        }
    }
    num_temporal_layer_switching_point_flags
    for( i = 0; i < num_temporal_layer_switching_point_flags; i++ )
        temporal_layer_switching_point_flag[i]
    num_ref_idx_l0_default_active_minus1
    num_ref_idx_l1_default_active_minus1
    pic_init_qp_minus26  /* relative to 26 */
    constrained_intra_pred_flag
    slice_granularity
    shared_pps_info_enabled_flag
    if( shared_pps_info_enabled_flag )
        if( adaptive_loop_filter_enabled_flag )
            alf_param( )
    if( cu_qp_delta_enabled_flag )
        max_cu_qp_delta_depth
    rbsp_trailing_bits( )
}
```

Listing (5)

[0171] A **wrap_indicator_flag** equal to 1 specifies that the POC numbering has transitioned the first time from one [0,···, *MaxPOC*-1] picture set to next. A **wrap_indicator_flag** equal to 0 specifies otherwise. In some configurations, the **wrap_indicator_flag** equal to 1 specifies that the POC numbering transitions from one [0,···, *MaxPOC*-1] picture set to another. A **wrap_indicator_flag** equal to 0 specifies otherwise.

[0172] In some configurations, a **wrap_indicator_flag** equal to 1 specifies that the poc_cycle has transitioned the first time for the next [0,···, *MaxPOC* -1] picture set. A **wrap_indicator_flag** equal to 0 specifies otherwise.

[0173] In some configurations, a **wrap_indicator_flag** equal to 1 specifies that the poc_cycle has transitioned from one [0, ···, *MaxPOC* -1] picture set to another. A **wrap_indicator_flag** equal to 0 specifies otherwise.

[0174] In some configurations, the **wrap_indicator_flag** may use more than one bit to identify additional information, for example, a direction of transition. In another configuration, the **wrap_indicator_flag** may be signaled as an unsigned integer that is variable length coded using entropy coding. In another configuration, the **wrap_indicator_flag** is a signed integer that is variable length coded using entropy coding.

[0175] seq_parameter_set_id identifies the sequence parameter set that is referred to by the picture parameter set.

The value of seq_parameter_set_id shall be in the range of 0 to 31, inclusive.

[0176] pic_parameter_set_id identifies the picture parameter set that is referred to in the slice header. The value of pic_parameter_set_id shall be in the range of 0 to 255, inclusive. entropy_coding_mode_flag selects the entropy decoding method to be applied for the syntax elements.

[0177] num_temporal_layer_switching_point_flags specifies how many temporal switching point flags are present. If temporal_id_nesting_flag is equal to 1, num_temporal_layer_switching_point_flags shall be equal to 0. temporal_layer_switching_point_flag[i] specifies if the current access point is a temporal switching point that allows decoding of higher temporal id layers following this access unit. If temporal_id_nesting_flag is equal to 1, temporal_layer_switching_point_flag[i] shall be inferred to be equal to 1. If temporal_id_nesting_flag is equal to 0 and num_temporal_layer_switching_point_flags is less than i, temporal_layer_switching_point_flag[i] shall be inferred to be equal to 0. It should be noted that when starting to decode a higher temporal layer i, the availability of required reference pictures can be guaranteed immediately following an IDR, or a picture with the temporal_id value j lower than i and temporal_switching_flag[j] equal to 1.

[0178] num_ref_idx_l0_default_active_minus1 specifies how num_ref_idx_l0_active_minus1 is inferred for P and B slices with num_ref_idx_active_override_flag equal to 0. The value of num_ref_idx_l0_default_active_minus1 shall be in the range of 0 to 31, inclusive.

[0179] num_ref_idx_l1_default_active_minus1 specifies how num_ref_idx_l1_active_minus1 is inferred for B slices with the num_ref_idx_active_override_flag equal to 0. The value of num_ref_idx_l1_default_active_minus1 shall be in the range of 0 to 31, inclusive.

[0180] pic_init_qp_minus26 specifies the initial value minus 26 of SliceQPY for each slice. The initial value is modified at the slice layer when a non-zero value of slice_qp_delta is decoded, and is modified further when a non-zero value of cu_qp_delta is decoded at the coding unit layer. The value of pic_init_qp_minus26 shall be in the range of -(26 + QpBdOffsetY) to +25, inclusive.

[0181] A constrained_intra_pred_flag equal to 0 specifies that intra prediction allows usage of residual data and decoded samples of neighboring macroblocks coded using inter macroblock prediction modes for the prediction of macroblocks coded using intra macroblock prediction modes. A constrained_intra_pred_flag equal to 1 specifies constrained intra prediction, in which case prediction of macroblocks coded using intra macroblock prediction modes only uses residual data and decoded samples from I macroblock types.

[0182] slice_granularity indicates the slice granularity within a picture. The value of slice_granularity shall not be larger than Min(Log2MaxCUSize - 4, log2_diff_max_min_coding_block_size). The variable SliceGranularity is set to the value of (slice_granularity << 1).

[0183] shared_pps_info_enabled_flag specifies that the shared information in the picture parameter set RBSP shall be used for the referred slices. If shared_pps_info_enabled_flag is equal to 1, the alf_param() in the picture parameter set RBSP shall be applied for the referred slices. Otherwise, the alf_param() in slice headers) shall be applied.

[0184] max_cu_qp_delta_depth specifies the maximum allowed depth that is used for specifying QPY values for a coding unit. The value of max_cu_qp_delta_depth shall be in the range of 0 to 15, inclusive.

[0185] The variable log2MinCUDQPSize specifies the minimum coding unit size that can further modify the value of QPY as follows: log2MinCUDQPSize = Log2MaxCUSize - max_cu_qp_delta_depth. alf_param() is a function that determines the adaptive loop filter parameter syntax. rbsp_trailing_bits() is a function that corresponds to a stop bit (equal to 1) followed by zero bits until byte alignment is achieved.

[0186] Listing (6) below illustrates an example of a bitstream syntax modification required to signal the wrap indicator in the slice header:

```
/*  Slice header RBSP syntax
*/
slice_header_rbsp( ) {
    lightweight_slice_flag
    if( !lightweight_slice_flag ) {
        slice_type
        pic_parameter_set_id
        if( IdrPicFlag )
            idr_pic_id
        pic_order_cnt
        wrap_indicator_flag
        if(!IdrPicFlag )
            buffer_description( )
        if(slice_type  = =  P  | |  slice_type  = =  B ) {
            num_ref_idx_active_override_flag
            if( num_ref_idx_active_override_flag ) {
                num_ref_idx_l0_active_minus1
                if( slice_type  = =  B )
                    num_ref_idx_l1_active_minus1
            }
        }
        ref_pic_list_modification( )
        ref_pic_list_combination( )
        if( IdrPicFlag )
            no_output_of_prior_pics_flag
    }
    if( entropy_coding_mode_flag  &&  slice_type  !=  I)
        cabac_init_idc
    first_slice_in_pic_flag
```

```
 if( first_slice_in_pic_flag == 0 )
     slice_address
 if( !lightweight_slice_flag ) {
     slice_qp_delta
     if( sample_adaptive_offset_enabled_flag )
         sao_param()
     if( deblocking_filter_control_present_flag ) {
         disable_deblocking_filter_idc
         if( disable_deblocking_filter_idc != 1 ) {
             slice_alpha_c0_offset_div2
             slice_beta_offset_div2
         }
     }
     if( slice_type = = B )
         collocated_from_l0_flag
     if( adaptive_loop_filter_enabled_flag ) {
         if( !shared_pps_info_enabled_flag )
             alf_param( )
         alf_cu_control_param( )
     }
 }
}
```

Listing (6)

[0187] A lightweight_slice_flag equal to 1 specifies that the value of slice header syntax elements not present shall be inferred to be equal to the value of slice header syntax elements in a proceeding slice. A lightweight_slice_flag equal to 0 specifies that the value of slice header syntax element values are transmitted in the current slice header. slice_type specifies the coding type of the slice P, B or I.

[0188] idr_pic_id identifies an instantaneous decoding refresh (IDR) picture. The values of idr_pic_id in all the slices of an IDR picture shall remain unchanged. When two consecutive access units in decoding order are both IDR access units, the value of idr_pic_id in the slices of the first such IDR access unit shall differ from the idr_pic_id in the second such IDR access unit. The value of idr_pic_id shall be in the range of 0 to 65535, inclusive.

[0189] pic_order_cnt specifies the picture order count of a coded picture and is used as an identifier in a buffer description application process and reference picture list creation processes. The pic_order_cnt syntax element shall be represented by log2_max_pic_order_cnt_minus4 + 4 bits. The value of pic_order_cnt shall be in the range of 0 to MaxPOC - 1, inclusive.

[0190] num_ref_idx_active_override_flag equal to 1 specifies that the syntax element num_ref_idx_l0_active_minus1 is present for P and B slices and that the syntax element num_ref_idx_l1_active_minus1 is present for B slices. num_ref_idx_active_override_flag equal to 0 specifies that the syntax elements num_ref_idxl0_active_minus1 and num_ref_idx_l1_active_minus1 are not present.

[0191] When the current slice is a P or B slice and field_pic_flag is equal to 0 and the value of num_ref_idx_l0_default_active_minus1 in the picture parameter set exceeds 15, num_ref_idx_active_override_flag shall be equal to 1. When the current slice is a B slice and field_pic_flag is equal to 0 and the value of num_ref_idx_l1_default_active_minus1 in the picture parameter set exceeds 15, num_ref_idx_active_override_flag shall be equal to 1. num_ref_idx_l0_active_minus1 specifies the maximum reference index for reference picture list 0 that shall be used to decode the slice.

[0192] When the current slice is a P or B slice and num_ref_idx_l0_active_minus1 is not present, num_ref_idx_l0_active_minus1 shall be inferred to be equal to num_ref_idx_l0_default_active_minus1. The range of num_ref_idx_l0_active_minus1 is specified as follows. If field_pic_flag is equal to 0, num_ref_idx_l0_active_minus1 shall be in the range of 0 to 15, inclusive. When MbaffFrameFlag is equal to 1, num_ref_idx_l0_active_minus1 is the maximum index value for the decoding of frame macroblocks and 2 * num_ref_idx_l0_active_minus1 + 1 is the maximum index value for the decoding of field macroblocks. Otherwise, (when field_pic_flag is equal to 1), num_ref_idx_l0_active_minus1 shall be in the range of 0 to 31, inclusive.

[0193] num_ref_idx_l1_active_minus1 specifies the maximum reference index for reference picture list 1 that shall be used to decode the slice. When the current slice is a B slice and num_ref_idx_l1_active_minus1 is not present, num_ref_idx_l1_active_minus1 shall be inferred to be equal to num_ref_idx_l1_default_active_minus1. The range of num_ref_idx_l1_active_minus1 is constrained as specified in the semantics for num_ref_idx_l0_active_minus1 with l0 and list 0 replaced by l1 and list 1, respectively.

[0194] no_output_of_prior_pics_flag specifies how the previously-decoded pictures in the decoded picture buffer are treated after decoding of an IDR picture. When the IDR picture is the first IDR picture in the bitstream, the value of no_output_of_prior_pics_flag has no effect on the decoding process. When the IDR picture is not the first IDR picture in the bitstream and the value of PicWidthInMbs, FrameHeightInMbs, or max_dec_frame_buffering derived from the active sequence parameter set is different from the value of PicWidthInMbs, FrameHeightInMbs, or max_dec_frame_buffering derived from the sequence parameter set active for the preceding picture, no_output_of_prior_pics_flag equal to 1 may (but should not) be inferred by the decoder, regardless of the actual value of no_output_of_prior_pics_flag.

[0195] cabac_init_idc specifies the index for determining the initialization table used in the initialization process for context variables. The value of cabac_init_idc shall be in the range of 0 to 2, inclusive.

[0196] first_slice_in_pic_flag indicates whether the slice is the first slice of the picture. If first_slice_in_pic_flag is equal to 1, the variables SliceAddress and LCUAddress are both set to 0 and the decoding starts with the first largest coding unit (LCU) in the picture.

[0197] slice_address specifies the address in slice granularity resolution in which the slice starts and shall be represented by (Ceil(Log2(NumLCUsInPicture)) + SliceGranularity) bits in the bitstream where NumLCUsInPicture is the number of LCUs in a picture. The variable LCUAddress is set to (slice_address >> SliceGranularity) and represents the LCU part of the slice address in raster scan order. The variable GranularityAddress is set to (slice_address - (LCUAddress << SliceGranularity)) and represents the sub-LCU part of the slice address expressed in z-scan order. The variable SliceAddress is then set to (LCUAddress << (log2_diff_max_min_coding_block_size << 1)) + (GranularityAddress << ((log2_diff_max_min_coding_block_size << 1) - SliceGranularity) and the slice decoding starts with the largest coding unit possible at the slice starting coordinate.

[0198] slice_qp_delta specifies the initial value of the luma quantization parameter QPY to be used for all the macroblocks in the slice until modified by the value of cu_qp_delta in the coding unit layer. The initial QPY quantization parameter for the slice is computed as: SliceQPY = 26 + pic_init_qp_minus26 + slice_qp_delta. The value of slice_qp_delta shall be limited such that SliceQPY is in the range of -QpBdOffsetY to +51, inclusive. "QpBdOffsetY = 6 * bit_depth_luma_minus8" specifies the value of the luma quantization parameter range offset. "bit_depth_luma_minus8 + 8" specifies the bit depth of the samples of the luma array. pic_init_qp_minus26 specifies the initial value minus 26 of SliceQPY for each slice.

[0199] A collocated_from_l0_flag equal to 1 specifies the picture that contains the co-located partition shall be derived from list 0, otherwise the picture shall be derived from list 1. ref_pic_list_modification() is a function that identifies the syntax of reference picture list modification. ref_pic_list_combination() is a function that identifies the syntax of reference picture list combination. alf_cu_control_param() is a function that identifies the syntax of adaptive loop filter coding unit control parameter. sao_param() is a function that identifies the syntax of sample adaptive offset parameter.

[0200] A disable_deblocking_filter_idc equal to 1 disables the application of the deblocking filter for certain block edges. A disable_deblocking_filter_idc equal to 0 enables the application of the deblocking filter for certain block edges. The deblocking process is controlled using values of syntax elements slice_alpha_c0_offset_div2 and slice_beta_offset_div2.

[0201] Listing (7) below illustrates an example of a bitstream syntax modification required to signal the wrap indicator in the Adaptation Parameter Set (APS):

```
/*  Adaptation parameter set RBSP syntax
*/
aps_rbsp( ) {
    aps_id
    wrap_indicator_flag
    aps_sample_adaptive_offset_flag
    aps_adaptive_loop_filter_flag
    if( aps_sample_adaptive_offset_flag || aps_adaptive_loop_filter_flag ) {
        aps_cabac_use_flag
        if( aps_cabac_use_flag ) {
            aps_cabac_init_idc
            aps_cabac_init_qp_minus26
        }
    }
    if( aps_adaptive_loop_filter_flag ) {
        alf_data_byte_count /* to enable skipping past data without parsing it */
    /*    byte_align()   this byte align happens between the non-CABAC and
    CABAC parts of the alf_param().  Once there is an all CABAC alf_param(),
    enable this byte_align() */
        alf_param( )
        byte_align()
     }
    if( aps_sample_adaptive_offset_flag ) {
        sao_data_byte_count /* to enable skipping past data without parsing it */
    byte_align ()
    sao_param( )
    /*    byte_align() this final byte align unnecessary as being taken care of by
    rbsp_trailing_bits() */
```

```
            rbsp_trailing_bits( )
        }
    }
```

Listing (7)

[0202] aps_id identifies the adaptation parameter set that is referred to in the slice header. An aps_sample_adaptive_offset_flag equal to 1 specifies that the SAO is on for slices referred to the current APS. When equal to 0, it specifies that the SAO is off for slices referred to the current APS. If there is no active APS, the aps_sample_adaptive_offset_flag value is inferred to be 0.

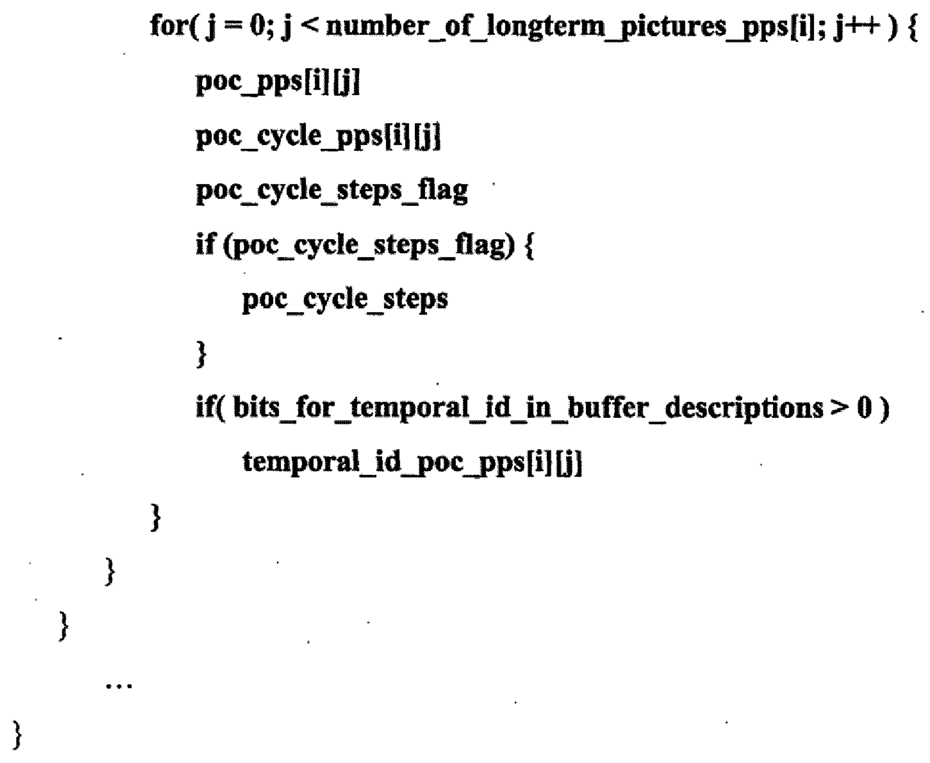
[0203] An aps_adaptive_loop_filter_flag equal to 1 specifies that the ALF is on for slices referred to the current APS. When equal to 0, it specifies that the ALF is off for slices referred to the current APS. If there is no active APS, the aps_adaptive_loop_filter_flag value is inferred to be 0.

[0204] An aps_cabac_use_flag equal to 1 specifies that the CABAC decoding process shall be used for sao_param() when present and used for alf_param() when present. When equal to 0, it specifies that the CAVLC decoding process shall be used for sao_param( ) when present and used for alf_param( ) when present.

[0205] aps_cabac_init_idc specifies the index for determining the initialization table used in the initialization process for context variables of SAO and ALF. The value of cabac_init_idc shall be in the range of 0 to 2, inclusive. aps_cabac_init_qp_minus26 specifies a quantization parameter minus 26, where the quantization parameter is used in the initialization process for context variables of SAO and ALF. alf_data_byte_count specifies a number of bytes. sao_data_byte_point specifies a number of bytes. byte_align() inserts 0 to 7 bits until alignment is reached.

[0206] In some configurations, the picture parameter set may include multiple buffer descriptions that correspond to long term pictures only. Listing (8) below illustrates an example of a bitstream syntax modification for signaling multiple buffer descriptions corresponding to long term pictures only.

```
/*  Picture parameter set RBSP syntax
*/
pic_parameter_set_rbsp( ) {
    ...
    bits_for_temporal_id_in_buffer_descriptions
    positive_pictures_in_buffer_descriptions_flag
    number_of_bds
    if( number_of_bds > 0 ) {
        for(i = 0; i < number_of_bds; i++){
            longterm_pictures_only_flag_pps[i]
            if (longterm_pictures_only_flag_pps[i]==0) {
                number_of_negative_pictures_pps[i]
                for( j = 0; j < number_of_negative_pictures_pps[i]; j++ ) {
                    negative_delta_poc_minus_one_pps[i][j]
                    if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                        temporal_id_negative_pps[i][j]
                }
                if( positive_pictures_in_buffer_descriptions_flag ){
                    number_of_positive_pictures_pps[i]
                    for( j = 0; j < number_of_positive_pictures_pps[i]; j++ ) {
                        delta_poc_minus_one_pps[i][j]
                        if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                            temporal_id_positive_pps[i][j]
                    }
                }
            }
            number_of_longterm_pictures_pps[i]
```



```
            for( j = 0; j < number_of_longterm_pictures_pps[i]; j++ ) {
                poc_pps[i][j]
                poc_cycle_pps[i][j]
                poc_cycle_steps_flag
                if (poc_cycle_steps_flag) {
                    poc_cycle_steps
                }
                if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                    temporal_id_poc_pps[i][j]
            }
        }
    }
    ...
}
```

Listing (8)

[0207] A **longterm_pictures_only_flag_pps[i]** equal to 1 indicates that the i-th buffer description contains only long term picture references. A **longterm_pictures_only_flag_pps[i]** equal to 0 identifies that the i-th buffer description contains both deltaPOC and long term picture references.

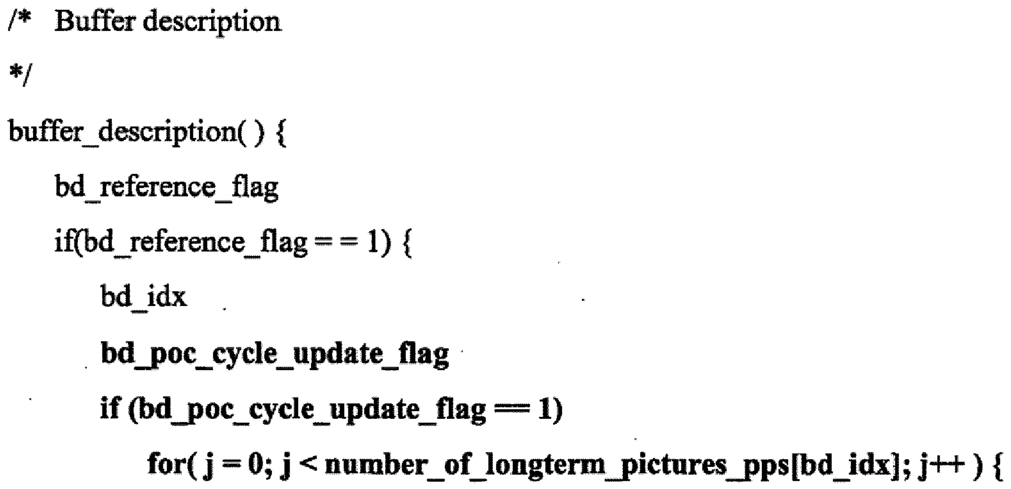
[0208] In some configurations, a buffer description may refer to a picture that is not available for reference. A picture that is not available for reference refers to a picture that has not been decoded prior to the current picture. Additionally, a picture that is not available for reference refers to a picture that has been decoded prior to the current picture but is identified as no longer available for reference. One example of a procedure for identifying that a picture is no longer available for reference is through the use of an IDR frame. An IDR frame denotes that the IDR frame and all pictures subsequently received do not refer to pictures received prior to receiving the IDR frame. Thus, all frames received prior to the IDR frame are no longer available for reference for the IDR frame and subsequently received frames.

[0209] In configurations where a buffer description refers to a picture that is not available for reference, a decoder does not allow the use of the picture that is not available for reference in predicting information used for decoding the current frame. In one configuration, this is done by not including the frame in the process of creating a set of reference frames. Thus, the output of the process of creating a set of reference frames may be the same if the buffer description refers to a picture that is not available for reference, if the buffer description does not refer to a picture that is not available for reference and if other pictures referred to by both buffer descriptions are the same. In another configuration, the picture that is not available for reference is included in the process of creating a set of reference frames. Thus, the output of the process of creating a set of reference frames is not the same if the buffer description refers to a picture that is not available for reference and if the buffer description does not refer to a picture that is not available for reference. In this configuration, a decoder does not allow selecting the picture that is not available for reference in predicting information used for decoding the current frame.

[0210] In some configurations, the i-th reference picture lists defined in the PPS: POCBD_pps[i], temporalIDBD_pps[i], POC_CYCLE_BD_pps[i] and the lists POCBD, temporalIDBD and POC_CYCLE_BD are partitioned based on if the reference picture is a past picture or a future picture, in display order, with respect to the current picture (being decoded). It should be noted that i may take on values from 0 to number_of_bds-1.

[0211] In some configurations, the temporal identifier may be replaced with a flag that signals or indicates whether the corresponding reference picture is used by the current picture (being decoded). As a result, the temporalID list temporalIDBBD and the i-th temporalID list in PPS temporalIDBD_pps[i] are replaced by lists usedByCurPicBD and usedByCurPicBD_pps[i], respectively, containing the above flag. In some configurations, the values of the flag may be used to partition the i-th reference picture lists defined in PPS: POCBD_pps[i], usedByCurPicBD_pps[i], POC_CYCLE_BD_pps[i] and the lists POCBD, usedByCurPicBD, POC_CYCLE_BD into one or more lists. In one configuration, entries with a zero value for the flag are assigned to one list and those with non-zero values (e.g., 1) are assigned to another list. It should be noted that i may take on values from 0 to number_of_bds-1.

[0212] Listing (9) illustrates another example of syntax modification for the PPS from AHG21. In particular, Listing (9) illustrates another example of a buffer description syntax used in slice headers as outlined in the AHG21 working draft. However, modifications to the syntax given in the AHG21 working draft in accordance with the systems and methods disclosed herein are denoted in bold text in Listing (9).



```
/*  Buffer description
*/
buffer_description( ) {
    bd_reference_flag
    if(bd_reference_flag = = 1) {
        bd_idx
        bd_poc_cycle_update_flag
        if (bd_poc_cycle_update_flag == 1)
            for( j = 0; j < number_of_longterm_pictures_pps[bd_idx]; j++ ) {
```

```
            poc_pps_override[j]
            poc_cycle_pps_override[j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_pps_override[j]
        }
    bd_poc_cycle_replace_count
    if (bd_poc_cycle_replace_count>0)
        for( j = 0; j < bd_poc_cycle_replace_count; j++ ) {
            bd_pps_replace_idx[j]
            poc_pps_replace[j]
            poc_cycle_pps_replace[j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_pps_replace[j]
        }

    bd_poc_cycle_delete_count
    if (bd_poc_cycle_delete_count >0 )
        for( j = 0; j < bd_poc_cycle_delete_count; j++ ) {
            bd_pps_delete_idx[j]
        }
    bd_poc_cycle_append_count
    if (bd_poc_cycle_append_count >0 )
        for( j = 0; j < bd_poc_cycle_append_count; j++ ) {
            poc_pps_append[j]
            poc_cycle_pps_append[j]
            if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                temporal_id_pps_append[j]
        }
} else {
    number_of_negative_pictures
    for( i = 0; i < number_of_negative_pictures; i++ ) {
    ...
```

```
            }
            if( positive_pictures_in_buffer_descriptions_flag ){
            ...
            }
            number_of_longterm_pictures
            for( j = 0; j < number_of_longterm_pictures; j++ ) {
                poc[j]
                poc_cycle[j]
                if( bits_for_temporal_id_in_buffer_descriptions > 0 )
                    temporal_id_poc[j]
            }
            if( number_of_negative_pictures + number_of_positive_pictures <
                 max_num_ref_frames ) {
                combine_with_reference_flag
                if( combine_with_reference_flag )
                    bd_combination_idx
            }
        }
    }
```

Listing (9)

[0213] Listing (9) contains many elements similar to that of Listing (4). Listing (9) also contains syntax modifications to the slice header to allow for selective replacement of reference pictures.

[0214] bd_poc_cycle_replace_count specifies the number of entries to be selectively replaced in the lists POCBD_pps[bd_idx], POC_CYCLE_BD_pps[bd_idx] and temporalIDbD_pps[bd_idx]. In some configurations, bd_poc_cycle_replace_count may be greater than or equal to 0. bd_pps_replace_idx[j] identifies the index of the entry to be replaced in the lists: POCBD_pps[bd_idx], POC_CYCLE_BD_pps[bd_idx] and temporalIDBD_pps[bd_idx].

[0215] POC_CYCLE_BD_pps[bd_idx][bd_pps_replace_idx[j]] of the referenced buffer description should be replaced for the current picture. In some configurations, future frames may also override poc_cycle information. poc_cycle_pps_replace[j] specifies the values to be used to replace the value within POC_CYCLE_BD_pps[bd_jdx][bd_pps_replace_idx[j]] for the current picture only.

[0216] poc_pps_replace[j] specifies the value to be used to replace the value within POCBD_pps[bd_idx][bd_pps_replace_idx[j]] for the current picture only. temporal_id_pps_replace[j] specifies the value to be used to replace the value within TemporalIDBD_pps[bd_idx][bd_pps_replace_idx[j]] for the current picture only.

[0217] In some configurations, if the i-th reference picture lists defined in PPS: POCBD_pps[i], usedByCurPicBD_pps[i], POC_CYCLE_BD_pps[i] are partitioned, then the selective replace, override, delete and append operations are first performed on a derived list generated using the partitioned list. After all or a subset of the operations are completed, the entries in the derived list may be partitioned back (e.g., restored) to the original lists from which the derived list was derived. It should be noted that i may take on values from 0 to number_of_bds-1.

[0218] In some configurations, the selective replace, override, delete and append operations are signaled separately for each list set belonging to a buffer description in the PPS (e.g., (negative deltaPOC list, negative deltaPOC usedByCurPicBD list) set, (positive deltaPOC list, positive deltaPOC usedByCurPicBD list) set and (poc list, poc_cycle list, poc and poc_cycle usedByCurPicBD list) set). In another configuration, a scaling parameter list may be present in each list set.

[0219] In some configurations, if a bitstream contains a first picture with a first resolution and a second picture with a second resolution, if the first resolution and the second resolution are not equal, and if the second picture is not an IDR picture or a Sequence Parameter Set (SPS) is not received after the first picture and prior to the second picture, then the decoded picture buffer may contain pictures having different resolutions. An encoder may then signal to the decoder that the reference pictures in the decoded picture buffer (DPB) are modified such that the reference picture comprises one of the following options:

(a) The first resolution,
(b) The second resolution, or
(c) Both the first and second resolutions

[0220] Some example use cases are given as follows. Option (a) is useful if a reference picture will only be used for motion prediction at the lower resolution. An example of option (a) is given as a first example 1653a illustrated in Figure 17.

[0221] Option (b) is useful when the bitstream contains a third picture that is subsequent to the second picture, where the third picture is at the first resolution and where the long term picture stored in the DPB may be used for motion prediction. An example of option (b) is given as a second example 1653b illustrated in Figure 17.

[0222] Option (c) is useful when a lower resolution version of the reference picture is used in motion prediction for lower resolution pictures. Additionally, when a higher resolution version of the reference picture is used for motion prediction when decoding a third picture that is subsequent to the second picture, where the third picture is at the first resolution. An example of option (c) is given as a third example 1653c in Figure 17.

[0223] The resolution switching may be signaled using the syntax modification as shown in Listing (10) below. The modifications in accordance with the systems and methods described herein are denoted in bold text below.

```
ref_pic_set( idx ) {
    num_negative_pics
    num_positive_pics
    number_of_longterm_pictures
    for( i = 0; i < num_negative_pics; i++ ) {
        delta_poc_s0_minus1[i]
        used_by_curr_pic_s0_flag[i]
        if ((refPicS0Width[i] != curPicWidth] || refPicS0Height[i] != curPicHeight) && adaptive_res_coding_flag)
            ref_pic_s0_resolution_idx[i]
    }
    for( i = 0; i < num_positive_pics; i++ ) {
        delta_poc_s1_minus1[i]
        used_by_curr_pic_s1_flag[i]
        if ((refPicS1Width[i] != curPicWidth || refPicS1Height[i] != curPicHeight) && adaptive_res_coding_flag)
            ref_pic_s1_resolution_idx[i]
    }
```

```
}
    for( i = 0; i < number_of_longterm_pictures; i++ ) {
        poc [i]
        poc_cycle[i]
        used_by_curr_pic_longterm_flag[i]
        if ((refPicS1Width[i] != curPicWidth || refPicS1Height[i] != curPicHeight)
  && adaptive_res_coding_flag)
            ref_pic_longterm_resolution_idx[i]
    }
```

Listing (10)

[0224] If an **adaptive_res_coding_flag** equals 0, then the sequence does not undergo resolution switching. If the **adaptive_res_coding_flag** equals 1, then the sequence undergoes resolution switching.

[0225] refPicS0Height[i] and refPicS0Width[i] represent the original height and width of the i-th reference picture corresponding to a negative deltaPOC. refPicS1Height[i] and refPicS1Width[i] represent the original height and width of the i-th reference picture corresponding to a positive deltaPOC. curPicHeight and curPicWidth represent the height and width of the current picture being decoded.

[0226] **num_negative_pics** specifies the number of the following delta_poc_s0_minus1[i] and used_by_curr_pic_s0_flag[i] syntax elements. **num_positive_pics** specifies the number of the following delta_poc_s1_minus1[i] and used_by_curr_pic_s1_flag[i] syntax elements.

[0227] **delta_poc_s0_minus1[i]** plus 1 specifies an absolute difference between two picture order count values. The value of delta_poc_s0_minus1[i] shall be in the range of 0 to 215 - 1, inclusive. delta_poc_s0_minus1[i] corresponds to negative deltaPOC values.

[0228] A **used_by_curr_pic_s0_flag[i]** equal to 0 specifies that the i-th reference picture that has a picture order count less than that of the current picture is not used for reference by the current picture. **delta_poc_s1_minus1[i]** plus 1 specifies an absolute difference between two picture order count values. The value of delta_poc_s1_minus1[i] shall be in the range of 0 to 215 - 1, inclusive. delta_poc_s1_minus1[i] corresponds to positive deltaPOC values.

[0229] A **used_by_curr_pic_s1_flag[i]** equal to 0 specifies that the i-th reference picture that has a picture order count greater than that of the current picture is not used for reference by the current picture.

[0230] A **used_by_curr_pic_longterm_flag[i**] equal to 0 specifies that the i-th long term reference picture is not used for reference by the current picture.

[0231] A **ref_pic_s0_resolution_idx[i]** equal to 0 specifies that the reference picture is scaled to make its resolution match the resolution of the current picture being decoded. A **ref_pic_s0_resolution_idx[i]** equal to 1 specifies that the reference picture is kept at the original resolution of the reference picture. A **ref_pic_s0_resolution_idx[i**] equal to 2 specifies that the reference picture resolution is maintained at both its original resolution and at the resolution of the current picture. **ref_pic_s0_resolution_idx[i]** is inferred to be 0 when not present.

[0232] A **ref_pic_s1_resolution_idx[i]** equal to 0 specifies that the reference picture is scaled to make its resolution match the resolution of the current picture being decoded. A **ref_pic_s1_resolution_idx[i]** equal to 1 specifies that the reference picture is kept at the original resolution of the reference picture. A **ref_pic_s1_resolution_idx[i]** equal to 2 specifies that the reference picture resolution is maintained at both its original resolution and at the resolution of the current picture. **ref_pic_s1_resolution_idx[i]** is inferred to be 0 when not present.

[0233] In one configuration, if a bitstream contains a first picture with a first resolution and a second picture and fourth picture with a second resolution, where the first resolution and second resolution are not equal and where the second picture is not an IDR picture or a Sequence Parameter Set (SPS) is not received after the first picture and prior to the second picture, then the second picture may only reference one reference picture of the first resolution. In a second configuration, if a bitstream contains a first picture with a first resolution and a second picture with a second resolution, where the first resolution and second resolution are not equal and where the second picture is not an IDR picture or a Sequence Parameter Set (SPS) is not received after the first picture and prior to the second picture, then the fourth picture may only reference one reference picture of the first resolution that is not referenced by other pictures of the second resolution.

[0234] For example, if the fourth picture is preceded by other pictures of the same resolution, the fourth picture may reference one reference picture that is the first resolution in addition to the reference pictures used by the pictures preceding the fourth picture that are of the same resolution. In other words, when decoding a picture of the second resolution, only one reference picture with a first resolution may be used for reference when decoding a picture. Once a picture with a first resolution is used for reference when decoding a picture with a second resolution, the picture with a first resolution that is used for reference is considered to have a resolution of the second resolution for decoding of subsequent pictures.

[0235] Figure 13 is a flow diagram illustrating another more specific configuration of a method 1200 for tracking a reference picture with reduced overhead referencing based on a designated picture. This method 1200 may be another approach for tracking which picture is being referenced when POCs are reused. An electronic device 204 (e.g., decoder 202) may receive 1202 a bitstream. For example, the decoder 202 may receive 1202 a bitstream 214 that includes an encoded reference picture. In some configurations, the bitstream 214 may also include overhead information (e.g., PPS, buffer description information, parameters, reference picture designation or identifier, etc.).

[0236] The electronic device 204 may decode 1204 a portion of the bitstream to produce a decoded reference picture. For example, the decoder 202 may decode 1204 a portion of the bitstream 214 to produce a decoded reference picture that is stored in frame memory 264. It should be noted that one or more portions of the bitstream 214 may be decoded 404 to produce one or more decoded reference pictures.

[0237] The electronic device 204 may associate 1206 a cycle parameter with a decoded picture set that includes the decoded reference picture. For example, the electronic device 204 may associate 1206 a cycle parameter "poc_cycle" with a decoded picture set or each picture in a decoded picture set that includes the decoded reference picture. The cycle parameter "poc_cycle" is described in greater detail above.

[0238] The electronic device 204 may determine 1208 whether a transition has occurred between picture sets. For example, the transition may be determined 1208 by examining the POC of a current picture being decoded (e.g., *CurPOC*) and comparing it to the POC of the last picture that was decoded (e.g., *LastPOC*). For instance, if the POC of the current picture (e.g., *CurPOC*) being decoded is less than the POC of the last decoded picture (e.g., *LastPOC*) and *LastPOC* - *CurPOC* is greater than a threshold *TH _FWD,* then a transition from an earlier picture set to a later picture set may be determined 1208. However, if the POC of the current picture being decoded (e.g., *CurPOC*) is greater than the POC of the last picture that was decoded (e.g., *LastPOC*) and *CurPOC* - *LastPOC* is greater than a threshold *TH_BCKWD,* then a transition from a later picture set to an earlier picture set may be determined 1208. For all other cases, it may be determined 1208 that no transition has occurred. In some configurations, the thresholds may take on values *TH_FWD* = *TH_BCKWD* = *MaxPOC*/2.

[0239] If the electronic device 204 determines 1208 that a transition has occurred between two picture sets, the electronic device 204 may modify 1210 the cycle parameter. For example, the electronic device 204 may decrement cycle parameters for each picture or each set of pictures in the DPB when the transition is from an earlier picture set. In another example, the electronic device 204 may increment the cycle parameters for each picture or each set of pictures in the DPB when the transition is from a later picture set. Thus, an update of all reference picture cycle parameters may be carried out with respect to the picture being decoded. This update procedure (e.g., determining 1208 whether a transition has occurred between picture sets and possibly modifying 1210 the cycle parameter(s)) may be executed once for each picture being decoded.

[0240] One alternative definition of the cycle parameter "poc_cycle" may be that the poc_cycle for the picture (currently) being decoded is 0. Thus, the set of pictures that includes the picture currently being decoded may be 0.

[0241] The poc_cycle of any other picture, such as the reference picture, may be calculated as the *MaxPOCSetIndex* of the reference picture minus the *MaxPOCSetIndex* of the picture being decoded. For example, if the *MaxPOCSetIndex* of the picture being decoded is *n* and the reference picture has a *MaxPOCSetIndex* that is n-1, then the poc_cycle of the reference picture may be (*n*-1)-*n*=-1.

[0242] It should be noted that the poc_cycle for a reference picture may depend on the *MaxPOCSetIndex* distance between the reference picture and the picture being decoded. This can be determined implicitly by keeping track of transitions (e.g., determining 1208 whether a transition has occurred) between one picture set of [0,···, *MaxPOC*-1] and the other picture set [0,···, *MaxPOC* -1] at both the encoder 108 and decoder 102.

[0243] The electronic device 204 may decode 1212 a picture based on the decoded reference picture. For example, a portion of the bitstream 214 (other than the portion decoded 1204 to produce the decoded reference picture) may be decoded 1212 based on the decoded reference picture. For example, the decoded reference picture (that has been tracked in the DPB) may be provided to a motion compensation module 260 in order to generate an inter-frame prediction signal 268 based on an inter-frame prediction mechanism. The inter-frame prediction signal 268 may then be used to decode 1212 the picture. In some configurations or instances, one or more decoded reference pictures may be used to decode 1212 the picture.

[0244] Figure 14 is a flow diagram illustrating one configuration of a method 1300 for determining whether a transition has occurred between picture sets. For example, Figure 14 provides one example of determining 1208 whether a transition has occurred between picture sets as illustrated in Figure 13. The electronic device 204 may determine 1302 whether the POC of the current picture being decoded (denoted "*CurPOC*," for example) is less than the POC of the last decoded picture (denoted *"LastPOC*," for example). For instance, the electronic device 204 may compare a POC of a current picture being decoded (e.g., *CurPOC*) to a POC of a picture that was decoded last (e.g., *LastPOC*) to make this determination 1302.

[0245] If *CurPOC* < *LastPOC*, the electronic device 204 may determine 1308 whether *LastPOC* - *CurPOC* is greater than a threshold *TH_FWD*. If *LastPOC* - *CurPOC* is greater than a threshold *TH_FWD,* the electronic device 204 may determine 1308 that a transition from an earlier picture set to a later picture set has occurred. However, if *LastPOC* - *CurPOC* is not greater than *TH_FWD,* the electronic device 204 may determine 1308 that no transition has occurred.

[0246] If *CurPOC* is not less than *LastPOC*, then the electronic device 204 may determine 1304 whether *CurPOC* is greater than *LastPOC.* If the electronic device 204 determines 1304 that *CurPOC* is greater than *LastPOC*, then the electronic device 204 may determine 1306 whether *CurPOC* - *LastPOC* is greater than a threshold *TH_BCKWD.* If the electronic device determines 1306 that *CurPOC* - *LastPOC* is greater than a threshold *TH_BCKWD*, then the electronic device 204 may determine 1306 that a transition from a later picture set to an earlier picture set has occurred. If the electronic device determines 1306 that *CurPOC* - *LastPOC* is not greater than a threshold *TH_BCKWD,* then the electronic device 204 may determine 1306 that no transition has occurred.

[0247] If the electronic device 204 determines 1304 that *CurPOC* is not greater than *LastPOC*, the electronic device may determine 1304 that no transition has occurred. In some configurations, the thresholds may take on values *TH_FWD = TH_BCKWD = MaxPOC/2.*

[0248] Figure 15 is a flow diagram illustrating another more specific configuration of a method 1400 for tracking a reference picture with reduced overhead referencing based on a designated picture. This method 1400 may be one approach for tracking which picture is being referenced when POCs are reused. An electronic device 204 (e.g., decoder 202) may receive 1402 a bitstream 214. For example, the decoder 202 may receive 1402 a bitstream 214 that includes an encoded reference picture (and other encoded pictures, for instance). In some configurations, the bitstream 214 may include overhead information (e.g., PPS, buffer description information, parameters, reference picture designation or identifier, etc.).

[0249] The electronic device 204 may decode 1404 a portion of the bitstream 214 to produce a decoded reference picture. For example, the decoder 202 may decode 1404 a portion of the bitstream 214 to produce a decoded reference picture that is stored in frame memory 264. It should be noted that one or more portions of the bitstream 214 may be decoded 1404 to produce one or more decoded reference pictures.

[0250] The electronic device 204 may associate 1406 a cycle parameter with a decoded picture set that includes the decoded reference picture. For example, the electronic device 204 may associate 1406 a cycle parameter "poc_cycle" with a decoded picture set that includes the decoded reference picture.

[0251] The electronic device 204 may determine 1408 whether a transition has occurred between picture sets. For example, each time a decoder 102 decodes a predetermined number of pictures in a set of pictures, the decoder 102 or electronic device B 104b may determine 1408 that a transition has occurred between two picture sets. In another example, each time a decoder 102 detects a cycle in POC (e.g., restarting from a maximum value to a minimum value), the decoder 102 or electronic device B 104b may determine 1408 that a transition has occurred between two picture sets.

[0252] If the electronic device 204 determines 1408 that a transition has occurred between picture sets, the electronic device 204 may modify 1410 (e.g., decrement) the cycle parameter. For example, the electronic device 204 decrements cycle parameters for each picture or each set of pictures in the DPB. In another example, the electronic device 204 may increment the cycle parameter.

[0253] The electronic device 204 may decode 1412 a picture based on the decoded reference picture. For example, a portion of the bitstream 214 (other than the portion decoded 1404 to produce the decoded reference picture) may be decoded 1412 based on the reference picture. For instance, the decoded reference picture (that has been tracked in the DPB) may be provided to a motion compensation module 260 in order to generate an inter-frame prediction signal 268 based on an inter-frame prediction mechanism. The inter-frame prediction signal 268 may then be used to decode 1412 the picture. In some configurations or instances, one or more decoded reference pictures may be used to decode 1412 the picture.

[0254] Figure 16 illustrates various components that may be utilized in an electronic device 1504. The electronic device 1504 may be implemented as one or more of the electronic devices (e.g., electronic devices 104, 204) described previously.

[0255] The electronic device 1504 includes a processor 1517 that controls operation of the electronic device 1504.

The processor 1517 may also be referred to as a CPU. Memory 1511, which may include both read-only memory (ROM), random access memory (RAM) or any type of device that may store information, provides instructions 1513a (e.g., executable instructions) and data 1515a to the processor 1517. A portion of the memory 1511 may also include non-volatile random access memory (NVRAM). The memory 1511 may be in electronic communication with the processor 1517.

[0256] Instructions 1513b and data 1515b may also reside in the processor 1517. Instructions 1513b and/or data 1515b loaded into the processor 1517 may also include instructions 1513a and/or data 1515a from memory 1511 that were loaded for execution or processing by the processor 1517. The instructions 1513b may be executed by the processor 1517 to implement the systems and methods disclosed herein.

[0257] The electronic device 1504 may include one or more communication interfaces 1519 for communicating with other electronic devices. The communication interfaces 1519 may be based on wired communication technology, wireless communication technology, or both. Examples of communication interfaces 1519 include a serial port, a parallel port, a Universal Serial Bus (USB), an Ethernet adapter, an IEEE 1394 bus interface, a small computer system interface (SCSI) bus interface, an infrared (IR) communication port, a Bluetooth wireless communication adapter, a wireless transceiver in accordance with 3rd Generation Partnership Project (3GPP) specifications and so forth.

[0258] The electronic device 1504 may include one or more output devices 1523 and one or more input devices 1521. Examples of output devices 1523 include a speaker, printer, etc. One type of output device that may be included in an electronic device 1504 is a display device 1525. Display devices 1525 used with configurations disclosed herein may utilize any suitable image projection technology, such as a cathode ray tube (CRT), liquid crystal display (LCD), tight-emitting diode (LED), gas plasma, electroluminescence or the like. A display controller 1527 may be provided for converting data stored in the memory 1511 into text, graphics, and/or moving images (as appropriate) shown on the display 1525. Examples of input devices 1521 include a keyboard, mouse, microphone, remote control device, button, joystick, trackball, touchpad, touchscreen, lightpen, etc.

[0259] The various components of the electronic device 1504 are coupled together by a bus system 1529, which may include a power bus, a control signal bus and a status signal bus, in addition to a data bus. However, for the sake of clarity, the various buses are illustrated in Figure 16 as the bus system 1529. The electronic device 1504 illustrated in Figure 16 is a functional block diagram rather than a listing of specific components.

[0260] Figure 17 is a diagram illustrating examples of use cases in accordance with the systems and methods disclosed herein. In particular, three examples 1653a-c are illustrated as described above. These three examples may occur if a bitstream contains a first picture with a first resolution and a second picture with a second resolution, if the first resolution and the second resolution are not equal, and if the second picture is not an IDR picture or a Sequence Parameter Set (SPS) is not received after the first picture and prior to the second picture.

[0261] The first example 1653a illustrates pictures of a first resolution 1655a and pictures of a second resolution 1657a stored in a decoded picture buffer. In the first example 1653a, an encoder may signal to a decoder that the reference pictures in the decoded picture buffer (DPB) are modified such that a reference picture is at the first resolution. This may be useful if the reference picture will only be used for motion prediction at the lower resolution.

[0262] The second example 1653b illustrates pictures of a first resolution 1655b and pictures of a second resolution 1657b stored in a decoded picture buffer. In the second example 1653b, an encoder may signal to a decoder that the reference pictures in the decoded picture buffer (DPB) are modified such that a reference picture is at the second resolution. This may be useful when the bitstream contains a third picture that is subsequent to the second picture, where the third picture is at the first resolution and where the long term picture stored in the DPB may be used for motion prediction.

[0263] The third example 1653c illustrates pictures of a first resolution 1655c and pictures of a second resolution 1657c stored in a decoded picture buffer. In the third example 1653c, an encoder may signal to a decoder that the reference pictures in the decoded picture buffer (DPB) are modified such that a reference picture is at both the first and second resolutions. This may be useful when a higher resolution version of the reference picture is used for motion prediction when decoding a third picture that is subsequent to the second picture, where the third picture is at the first resolution.

[0264] The term "computer-readable medium" refers to any available medium that can be accessed by a computer or a processor. The term "computer-readable medium," as used herein, may denote a computer- and/or processor-readable medium that is non-transitory and tangible. By way of example, and not limitation, a computer-readable or processor-readable medium may comprise RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other medium that can be used to carry or store desired program code in the form of instructions or data structures and that can be accessed by a computer or processor. Disk and disc, as used herein, includes compact disc (CD), laser disc, optical disc, digital versatile disc (DVD), floppy disk and Blu-ray® disc where disks usually reproduce data magnetically, while discs reproduce data optically with lasers.

[0265] It should be noted that one or more of the methods described herein may be implemented in and/or performed using hardware. For example, one or more of the methods or approaches described herein may be implemented in and/or realized using a chipset, an application-specific integrated circuit (ASIC), a large-scale integrated circuit (LSI) or integrated circuit, etc.

[0266] Each of the methods disclosed herein comprises one or more steps or actions for achieving the described method. The method steps and/or actions may be interchanged with one another and/or combined into a single step without departing from the scope of the claims. In other words, unless a specific order of steps or actions is required for proper operation of the method that is being described, the order and/or use of specific steps and/or actions may be modified without departing from the scope of the claims.

[0267] It is to be understood that the claims are not limited to the precise configuration and components illustrated above. Various modifications, changes and variations may be made in the arrangement, operation and details of the systems, methods, and apparatus described herein without departing from the scope of the claims.

[0268] Although the present invention is defined in the attached claims, it should be understood that the present invention can also (alternatively) be defined in accordance with the following embodiments:

1. A method for decoding video bitstream comprising:

   receiving a reference description for a decoded reference pictures from said bitstream;
   determining a picture order count (POC) of a current picture for decoding;
   identifying said decoded reference pictures of said current picture from a decoded picture buffer, based on said picture order count and said reference description;
   decoding said current picture using inter-frame prediction based on said decoded reference pictures; and
   buffering said decoded picture in said decoded picture buffer to future prediction, wherein
   said reference description includes a modification parameter for long term reference pictures; and
   said identifying uses said modification parameter to modify at least a part of references for a set of decoded reference pictures.

2. The method of embodiment 1, wherein said modification parameter is a poc cycle parameter.

3. The method of embodiment 1, wherein said modification parameter is an indicator to start another poc cycle.

4. The method of embodiment 1 to embodiment 3, wherein said reference information includes a picture order count for said decoded reference pictures.

5. The method of embodiment 1, wherein said poc cycle parameter is in a slice header.

6. An electronic device configured for decoding video bitstream, comprising: a processor;

   memory in electronic communication with the processor;
   instruction stored in the memory, the instructions being executable to:

      receive a reference description for a decoded reference pictures from said bitstream;
      determine a picture order count (POC) of a current picture for decoding;
      identify said decoded reference pictures of said current picture from a decoded picture buffer, based on said picture order count and said reference description; decode said current pictures using inter-frame prediction based on said decoded reference pictures; and
      buffer said decoded pictures in said decoded picture buffer to future prediction, wherein
      said reference description includes a modification parameter for long term reference pictures; and
      said modification parameter is used to modify at least a part of references for a set of decoded reference pictures.

Claims

1. A video bitstream including:

   a reference picture;
   a buffer description of said reference picture, said buffer description including a picture order count for said reference picture; and
   a wrap indicator indicating a transition between two sets of pictures, the wrap indicator being a one bit flag, the wrap indicator being configured to determine a current picture order count cycle parameter including updating

a previous picture order count cycle parameter,
wherein the picture order count cycle parameter references sets of pictures including reference pictures.

2. The video bitstream of claim 1, not including the picture order count cycle parameters.

FIG. 1

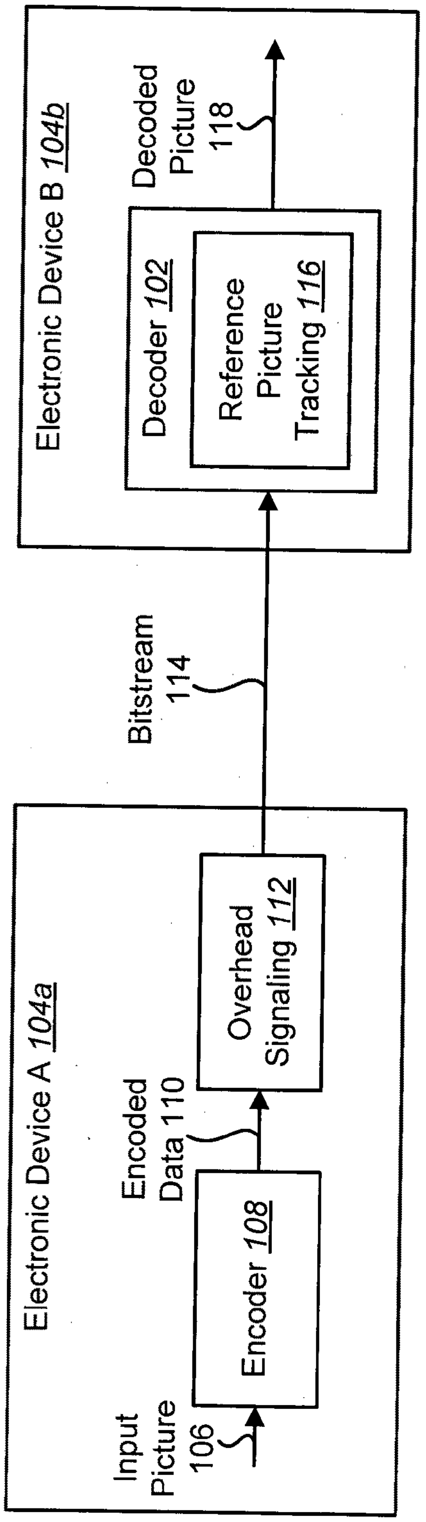
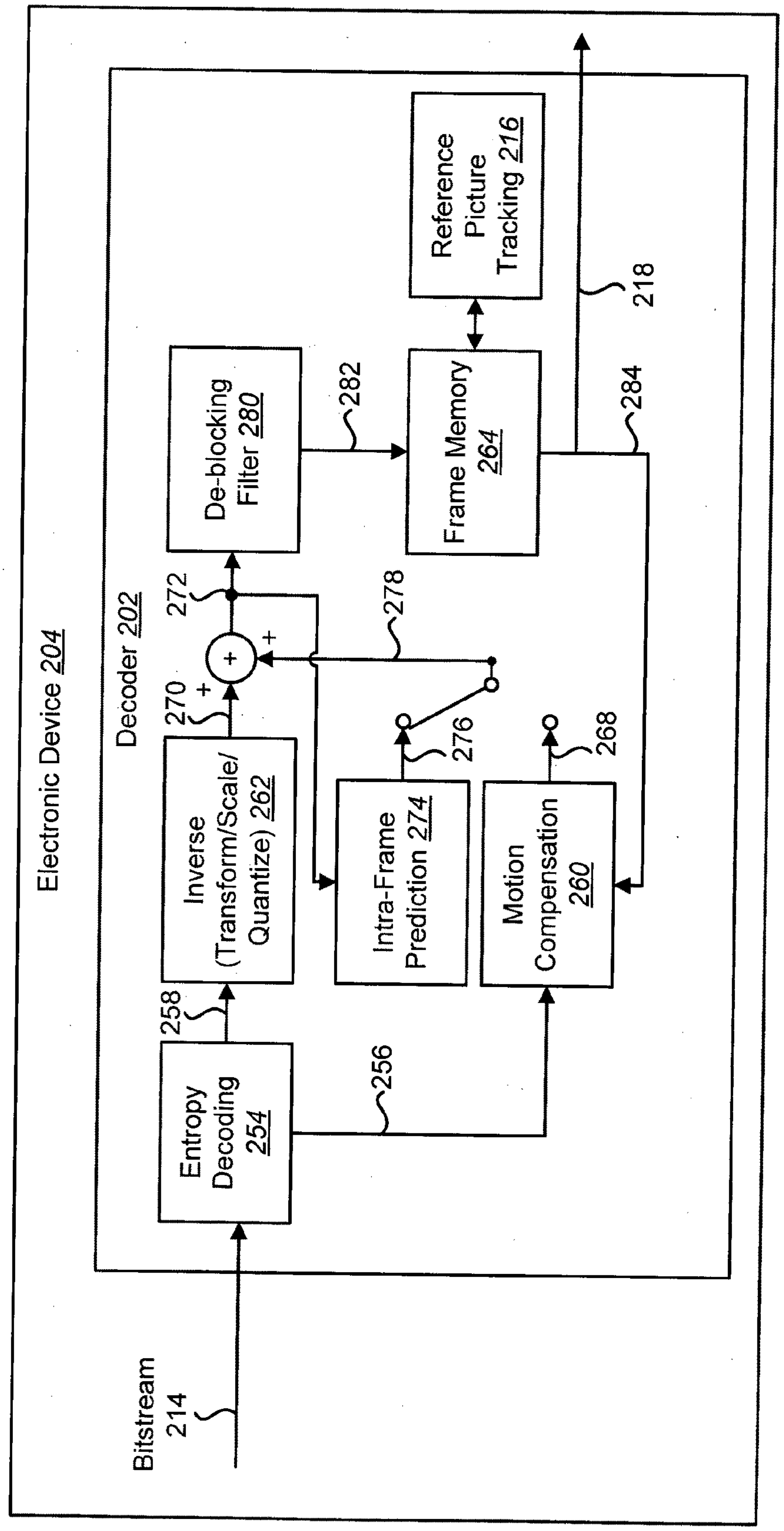

Electronic Device A 104a
Input Picture 106
Encoder 108
Encoded Data 110
Overhead Signaling 112
Bitstream 114
Electronic Device B 104b
Decoder 102
Reference Picture Tracking 116
Decoded Picture 118

FIG. 2


Electronic Device 204
Decoder 202
Bitstream 214
Entropy Decoding 254
258
256
Inverse (Transform/Scale/ Quantize) 262
270
+
+
272
278
De-blocking Filter 280
282
Frame Memory 264
284
Reference Picture Tracking 216
218
Intra-Frame Prediction 274
276
Motion Compensation 260
268

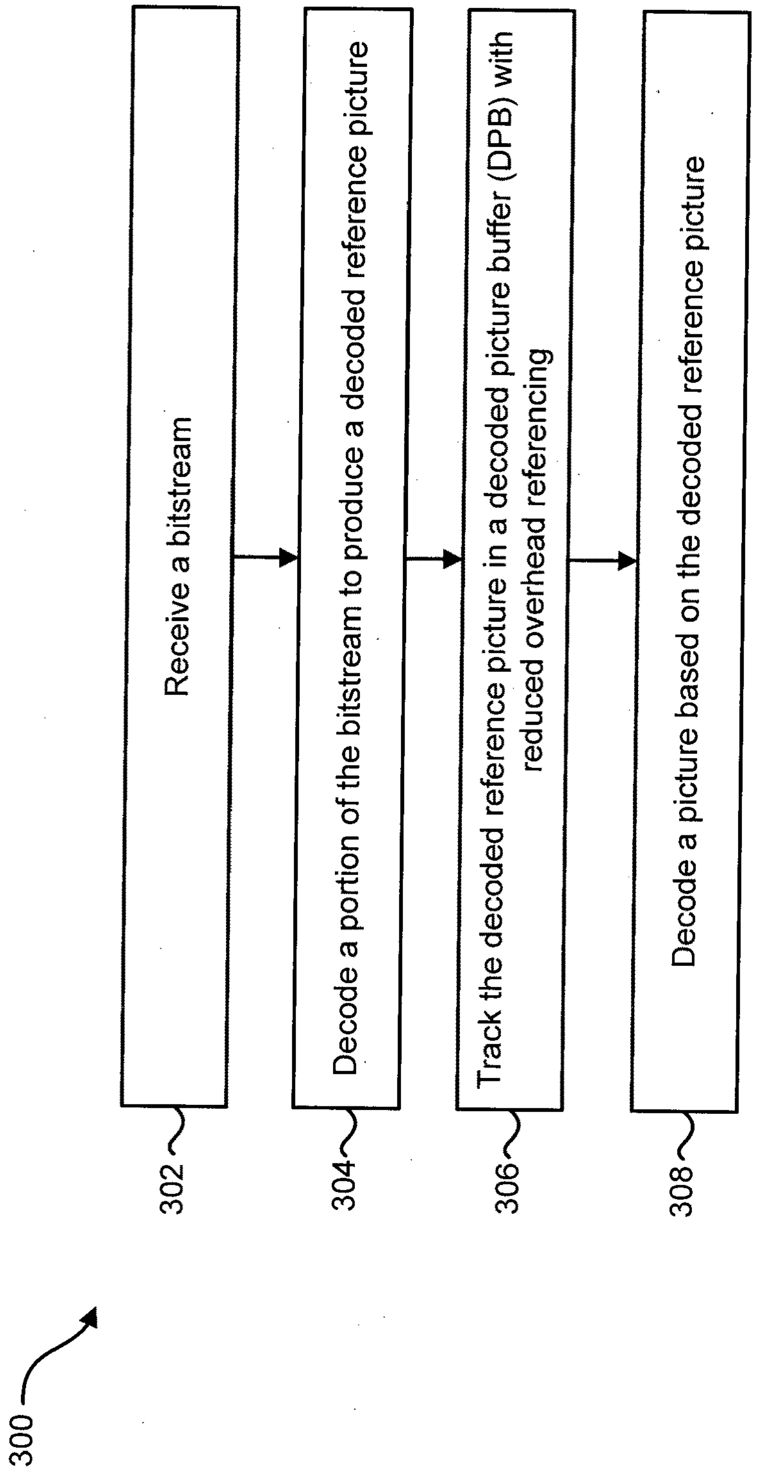

300
302
Receive a bitstream
304
Decode a portion of the bitstream to produce a decoded reference picture
306
Track the decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing
308
Decode a picture based on the decoded reference picture

FIG. 3

FIG. 4

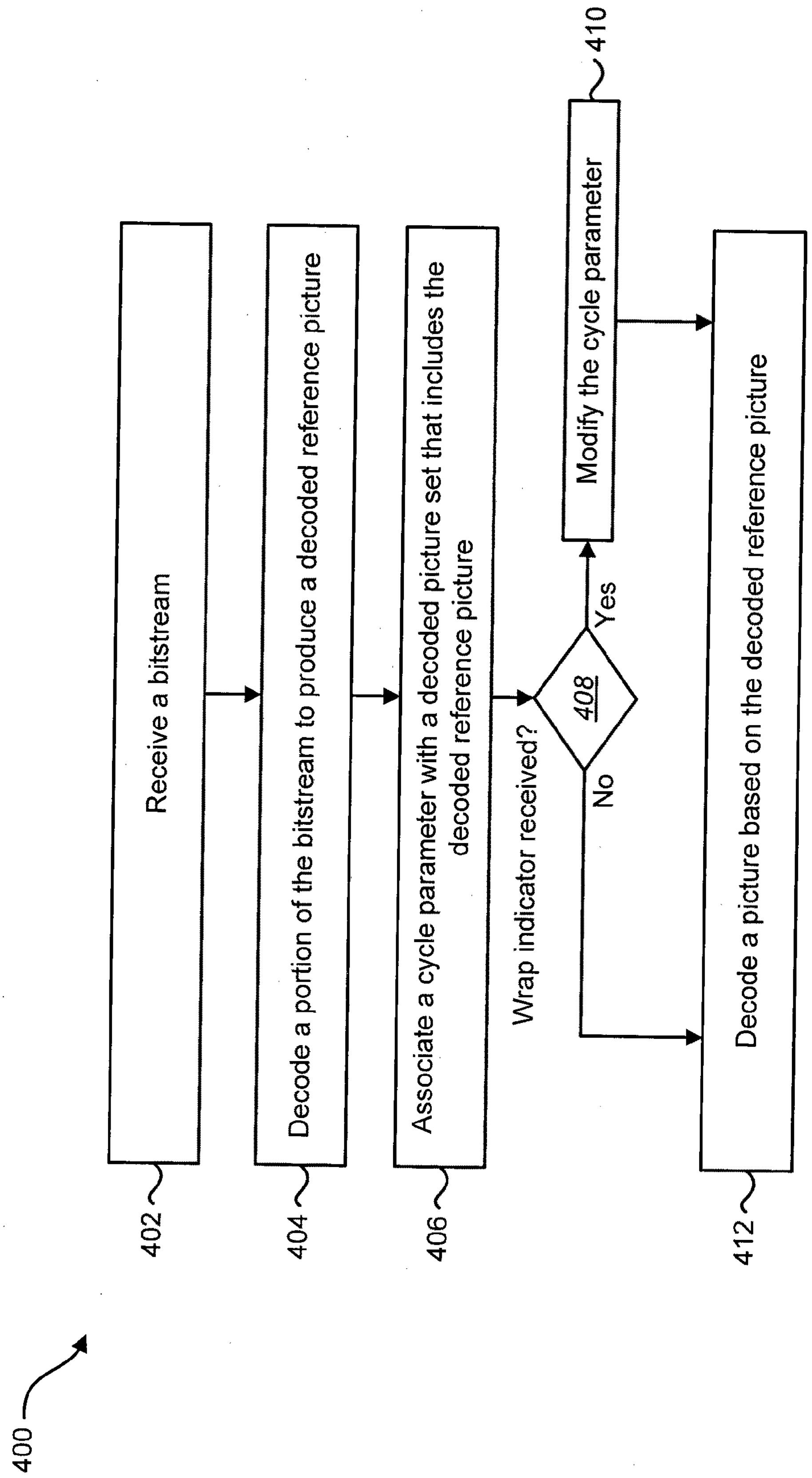

400
Receive a bitstream
402
Decode a portion of the bitstream to produce a decoded reference picture
404
Associate a cycle parameter with a decoded picture set that includes the decoded reference picture
406
Wrap indicator received?
408
Yes
No
Modify the cycle parameter
410
Decode a picture based on the decoded reference picture
412

FIG. 5

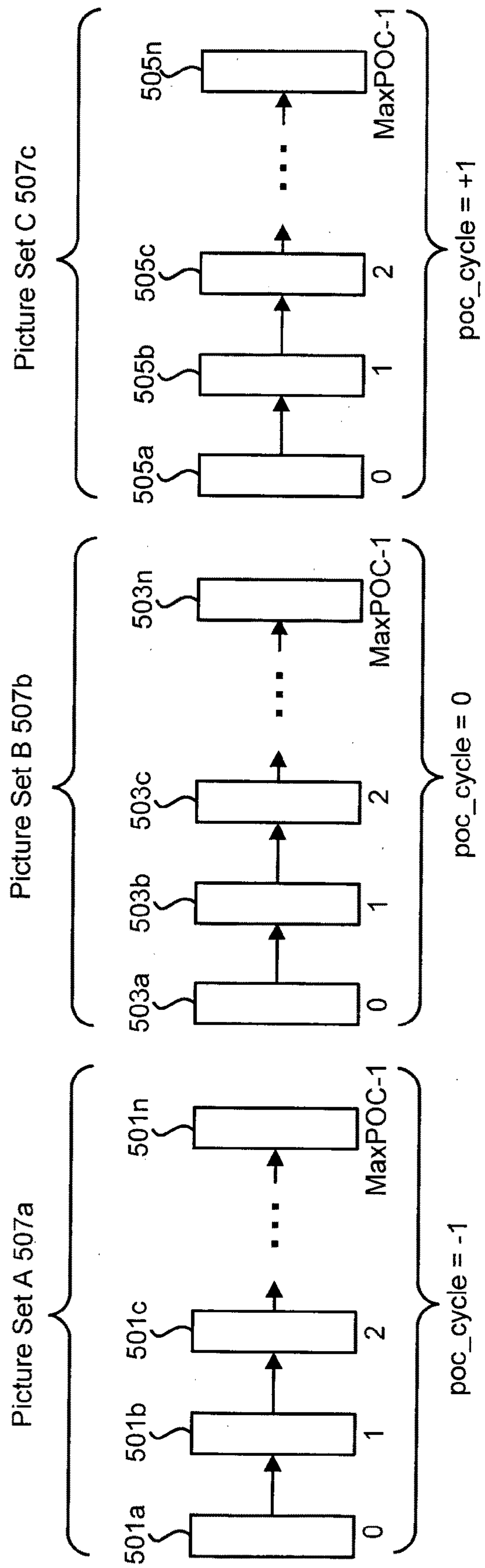

Picture Set A 507a
501a
501b
501c
501n
0
1
2
MaxPOC-1
poc_cycle = -1
Picture Set B 507b
503a
503b
503c
503n
0
1
2
MaxPOC-1
poc_cycle = 0
Picture Set C 507c
505a
505b
505c
505n
0
1
2
MaxPOC-1
poc_cycle = +1

FIG. 6

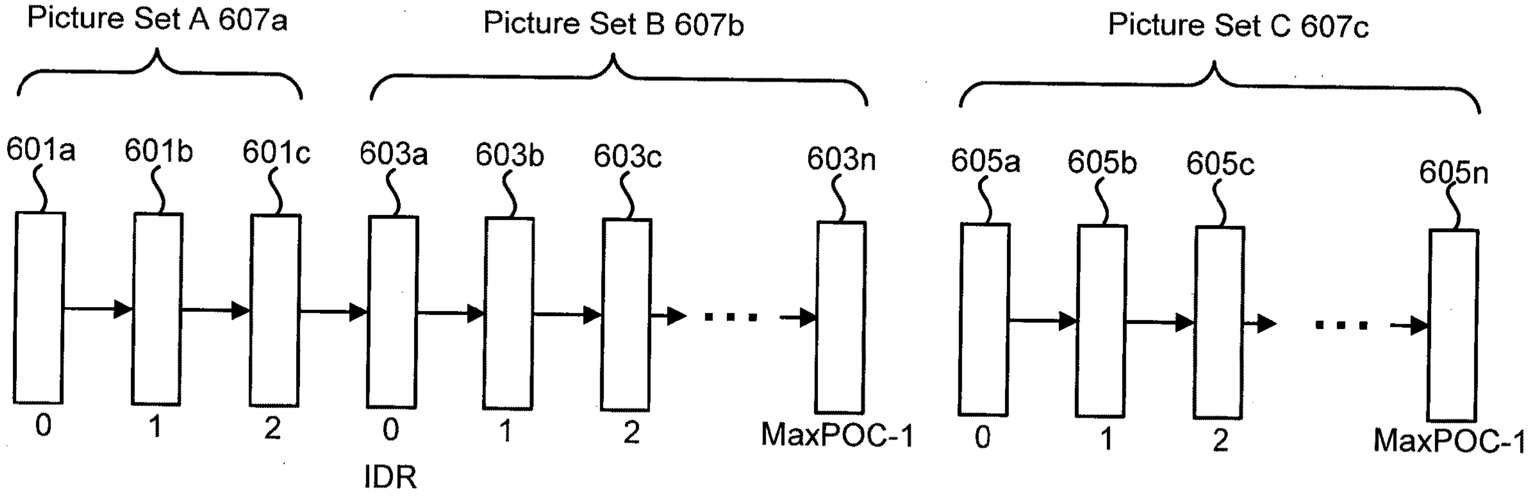

Picture Set A 607a
Picture Set B 607b
Picture Set C 607c
601a
601b
601c
603a
603b
603c
603n
605a
605b
605c
605n
0
1
2
0
IDR
1
2
MaxPOC-1
0
1
2
MaxPOC-1

FIG. 7

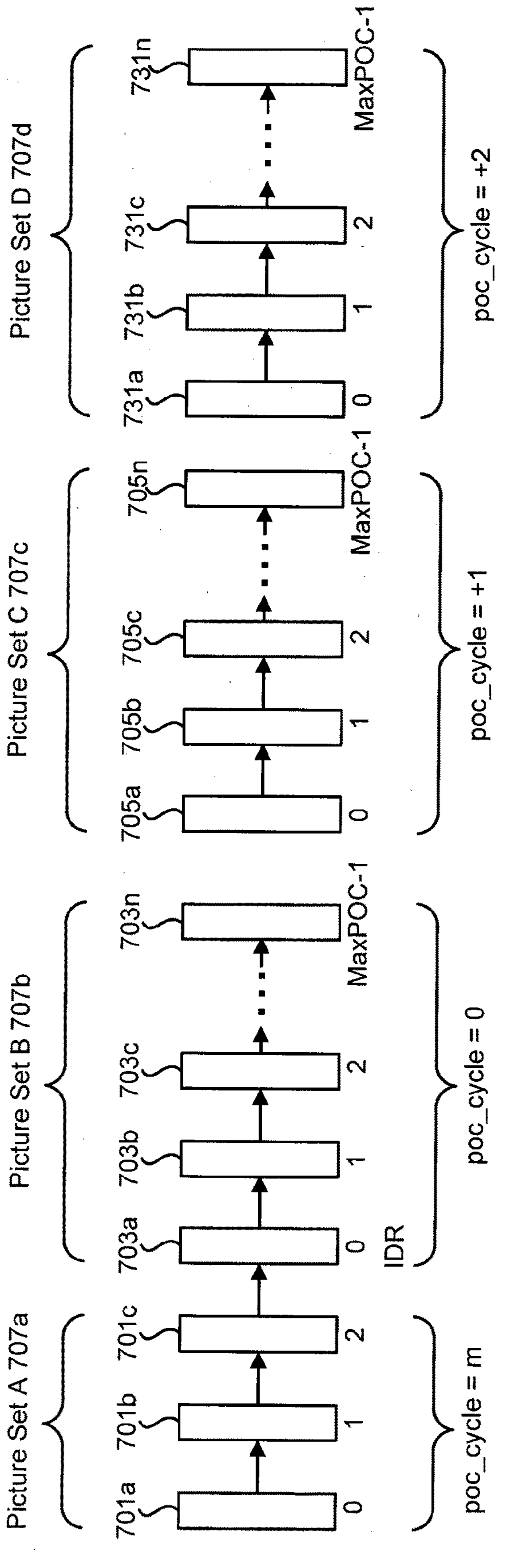
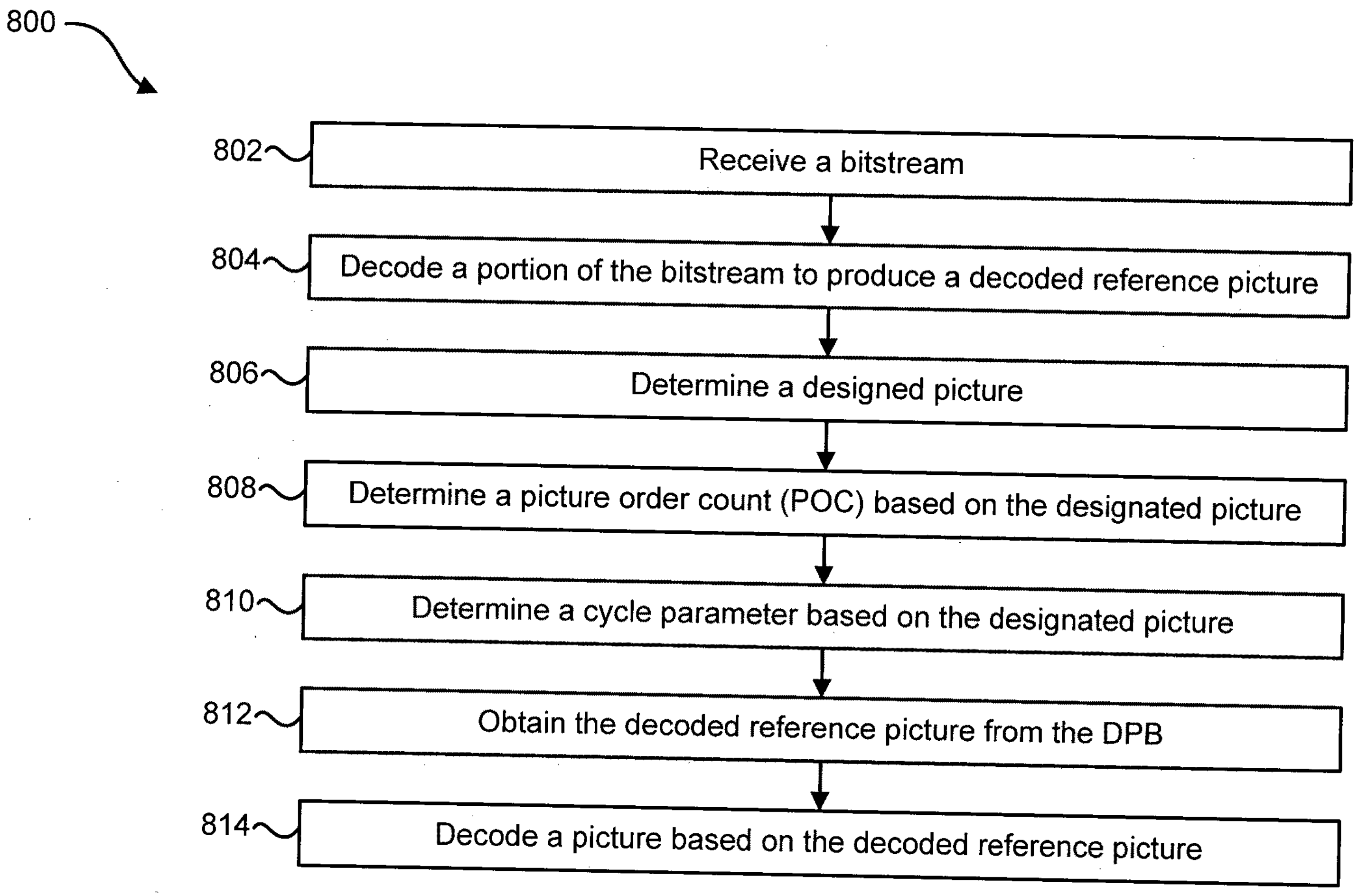

Picture Set A 707a
701a
701b
701c
0
1
2
poc_cycle = m
Picture Set B 707b
703a
703b
703c
703n
0
1
2
MaxPOC-1
IDR
poc_cycle = 0
Picture Set C 707c
705a
705b
705c
705n
0
1
2
MaxPOC-1
poc_cycle = +1
Picture Set D 707d
731a
731b
731c
731n
0
1
2
MaxPOC-1
poc_cycle = +2

FIG. 8


800
802
Receive a bitstream
804
Decode a portion of the bitstream to produce a decoded reference picture
806
Determine a designed picture
808
Determine a picture order count (POC) based on the designated picture
810
Determine a cycle parameter based on the designated picture
812
Obtain the decoded reference picture from the DPB
814
Decode a picture based on the decoded reference picture

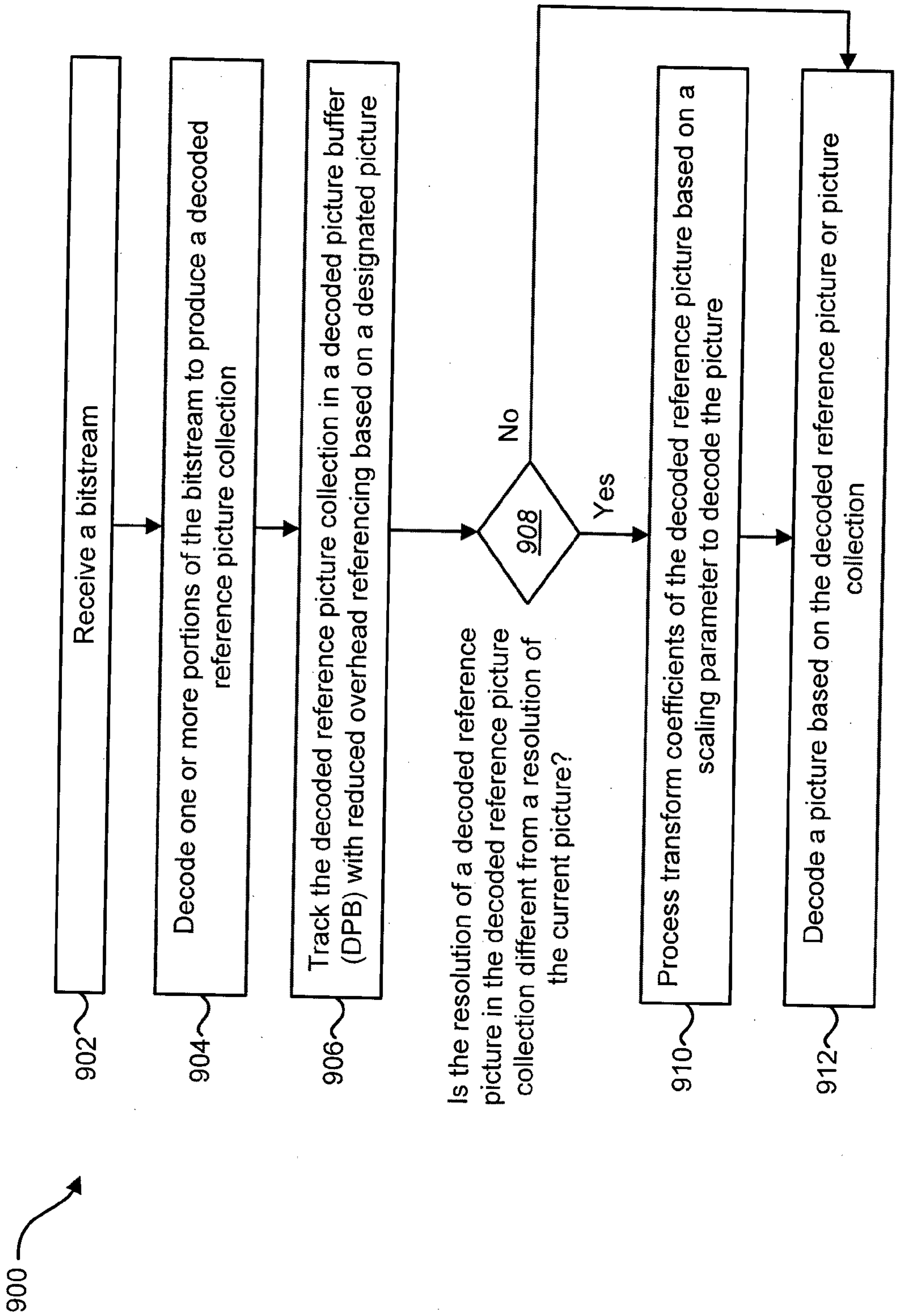

900
902
Receive a bitstream
904
Decode one or more portions of the bitstream to produce a decoded reference picture collection
906
Track the decoded reference picture collection in a decoded picture buffer (DPB) with reduced overhead referencing based on a designated picture
908
Is the resolution of a decoded reference picture in the decoded reference picture collection different from a resolution of the current picture?
No
Yes
910
Process transform coefficients of the decoded reference picture based on a scaling parameter to decode the picture
912
Decode a picture based on the decoded reference picture or picture collection

FIG. 9

FIG. 10

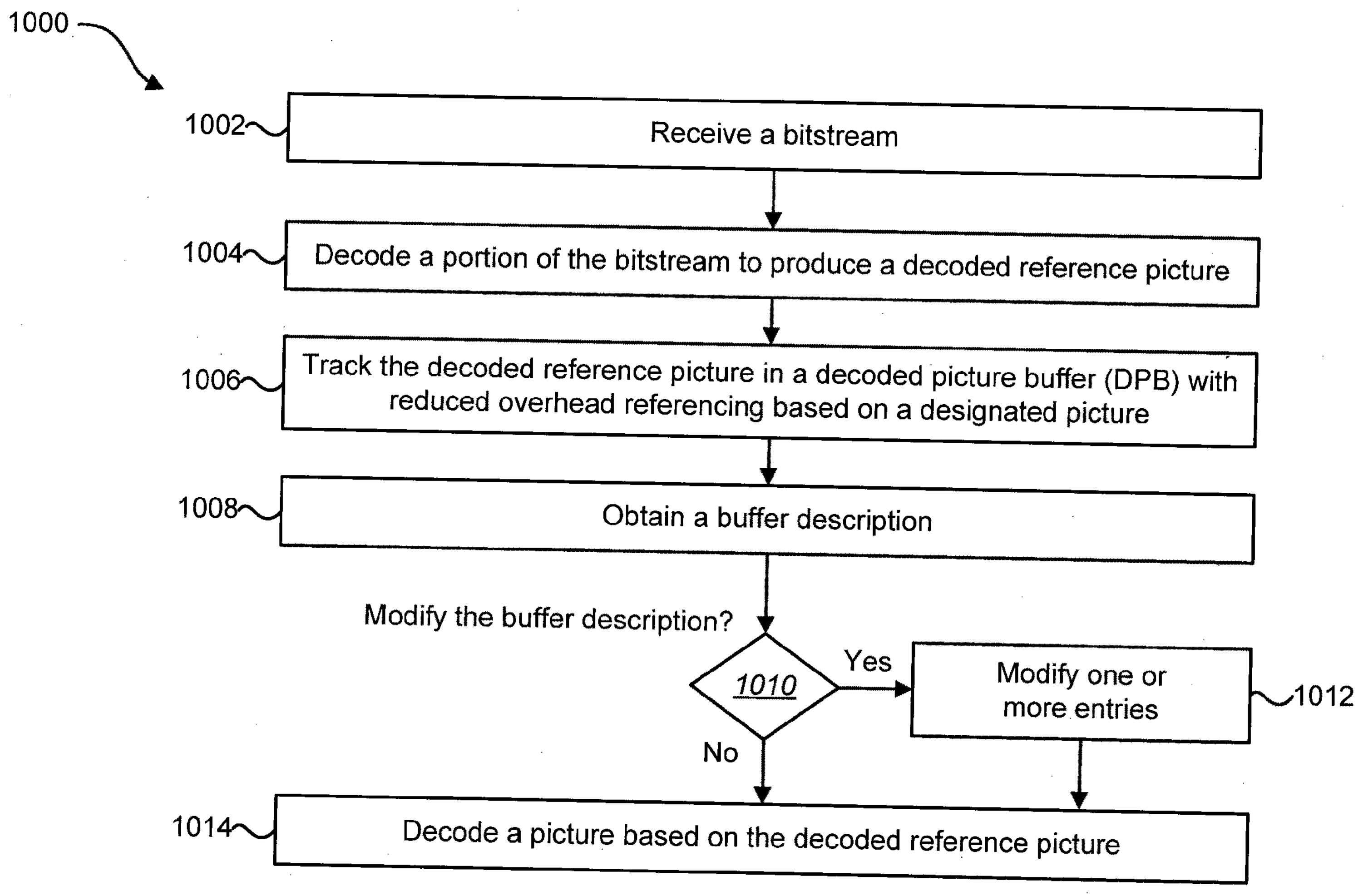

1000
1002
Receive a bitstream
1004
Decode a portion of the bitstream to produce a decoded reference picture
1006
Track the decoded reference picture in a decoded picture buffer (DPB) with reduced overhead referencing based on a designated picture
1008
Obtain a buffer description
Modify the buffer description?
1010
Yes
No
Modify one or more entries
1012
1014
Decode a picture based on the decoded reference picture

FIG. 11

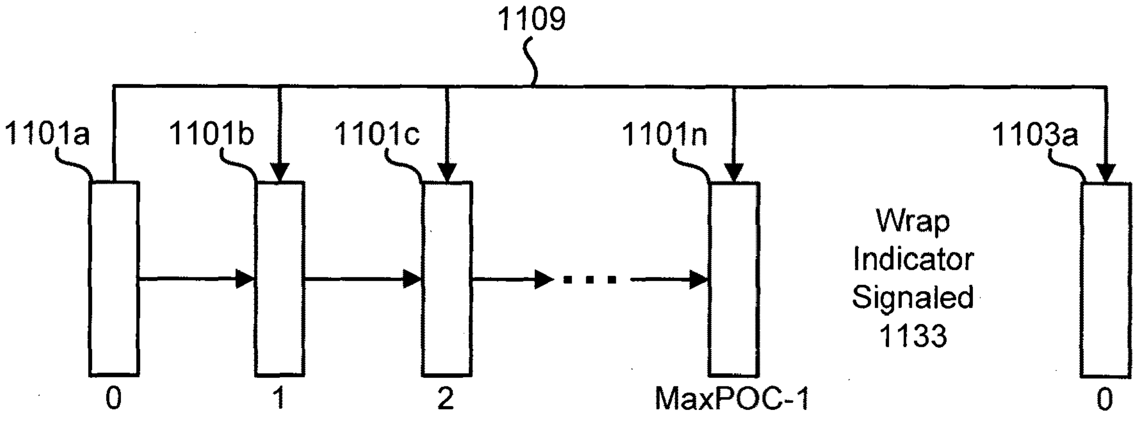

1109
1101a
1101b
1101c
1101n
1103a
Wrap
Indicator
Signaled
1133
0
1
2
MaxPOC-1
0

FIG. 12

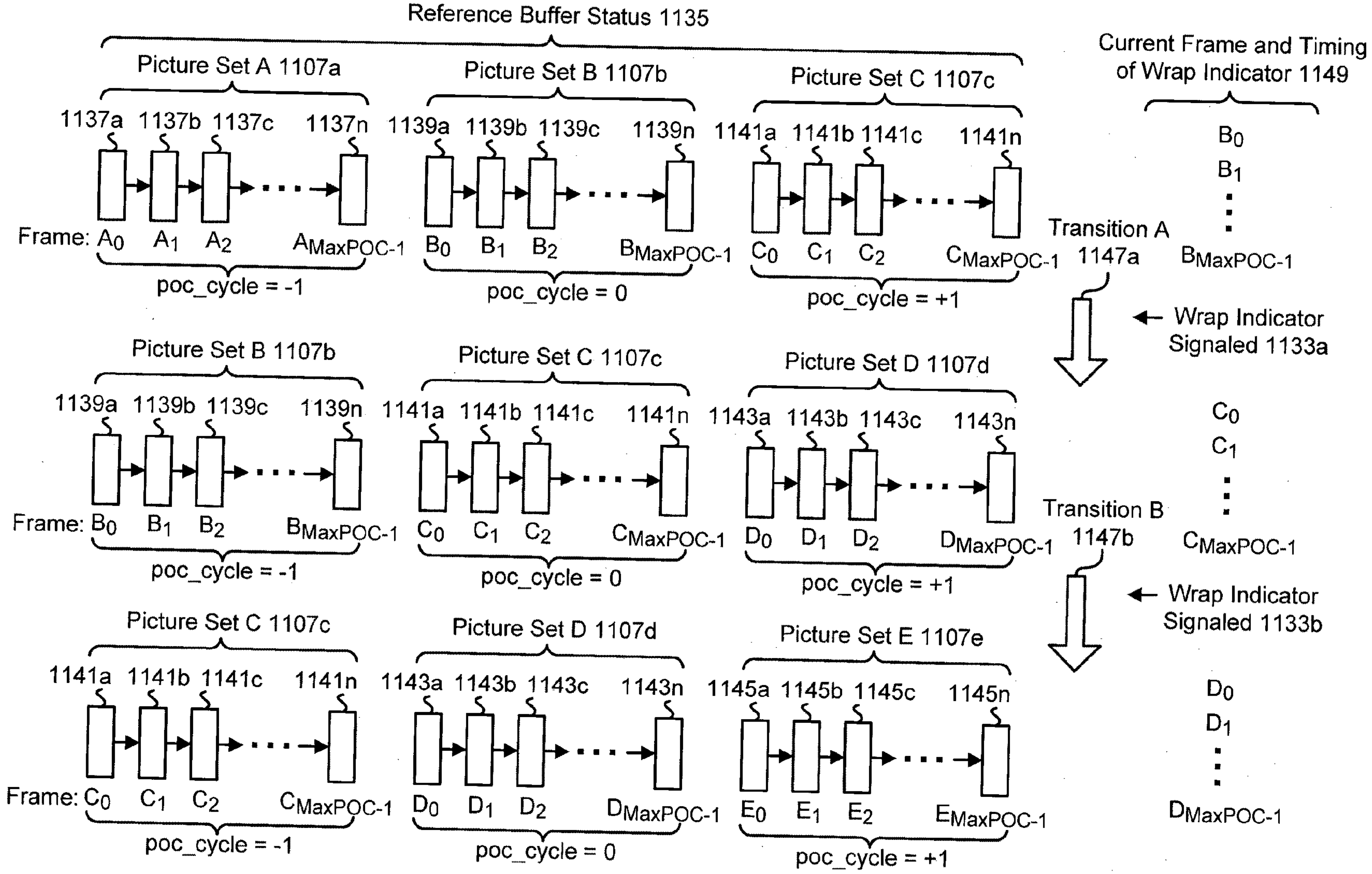

Reference Buffer Status 1135
Picture Set A 1107a
1137a 1137b 1137c 1137n
Frame: A_0 A_1 A_2 $A_{MaxPOC-1}$
poc_cycle = -1
Picture Set B 1107b
1139a 1139b 1139c 1139n
B_0 B_1 B_2 $B_{MaxPOC-1}$
poc_cycle = 0
Picture Set C 1107c
1141a 1141b 1141c 1141n
C_0 C_1 C_2 $C_{MaxPOC-1}$
poc_cycle = +1
Current Frame and Timing of Wrap Indicator 1149
B_0
B_1
$B_{MaxPOC-1}$
Transition A 1147a
Wrap Indicator Signaled 1133a
Picture Set B 1107b
1139a 1139b 1139c 1139n
Frame: B_0 B_1 B_2 $B_{MaxPOC-1}$
poc_cycle = -1
Picture Set C 1107c
1141a 1141b 1141c 1141n
C_0 C_1 C_2 $C_{MaxPOC-1}$
poc_cycle = 0
Picture Set D 1107d
1143a 1143b 1143c 1143n
D_0 D_1 D_2 $D_{MaxPOC-1}$
poc_cycle = +1
C_0
C_1
$C_{MaxPOC-1}$
Transition B 1147b
Wrap Indicator Signaled 1133b
Picture Set C 1107c
1141a 1141b 1141c 1141n
Frame: C_0 C_1 C_2 $C_{MaxPOC-1}$
poc_cycle = -1
Picture Set D 1107d
1143a 1143b 1143c 1143n
D_0 D_1 D_2 $D_{MaxPOC-1}$
poc_cycle = 0
Picture Set E 1107e
1145a 1145b 1145c 1145n
E_0 E_1 E_2 $E_{MaxPOC-1}$
poc_cycle = +1
D_0
D_1
$D_{MaxPOC-1}$

FIG. 13

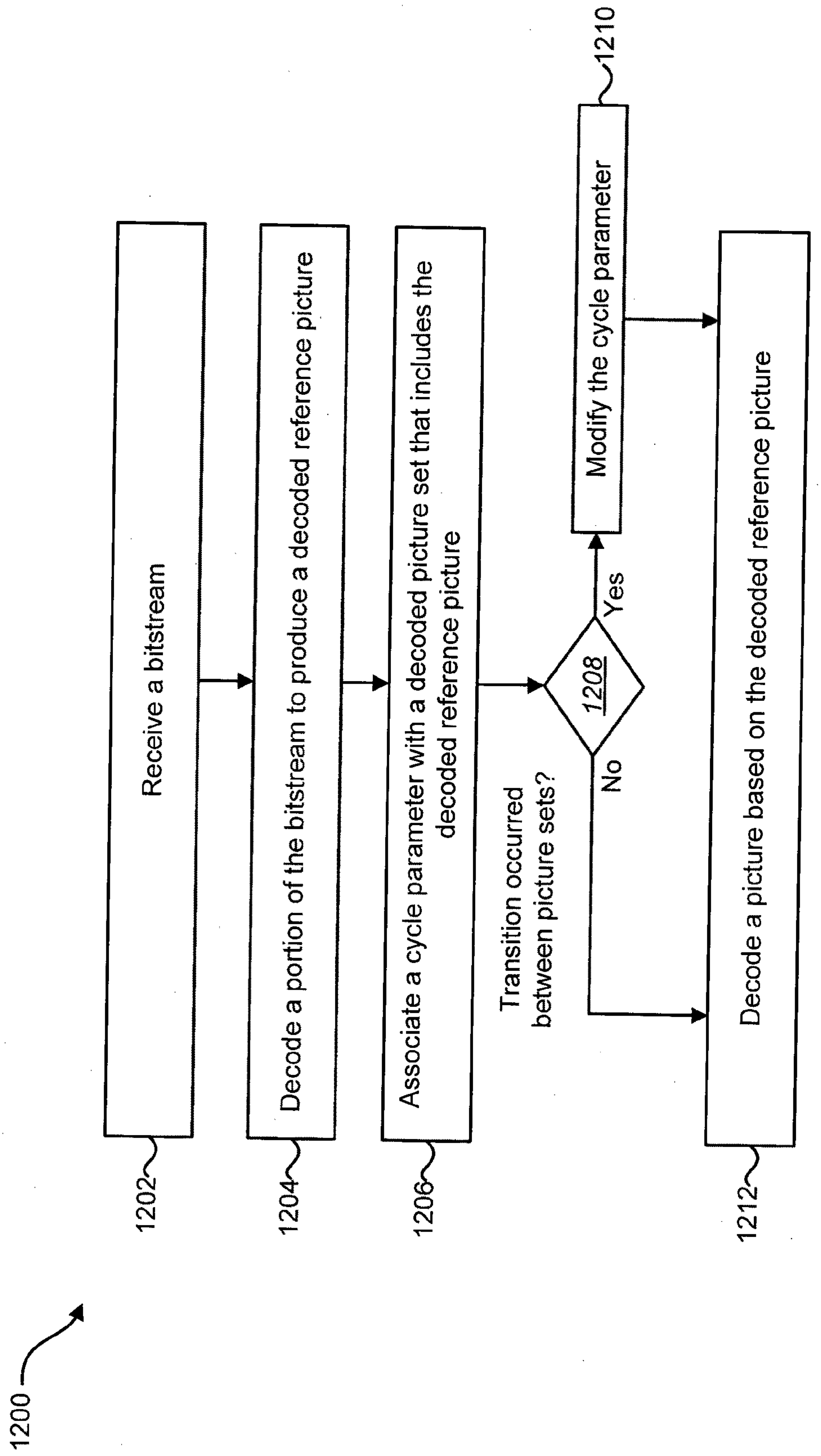

1200
Receive a bitstream
1202
Decode a portion of the bitstream to produce a decoded reference picture
1204
Associate a cycle parameter with a decoded picture set that includes the decoded reference picture
1206
Transition occurred between picture sets?
1208
Yes
No
Modify the cycle parameter
1210
Decode a picture based on the decoded reference picture
1212

FIG. 14

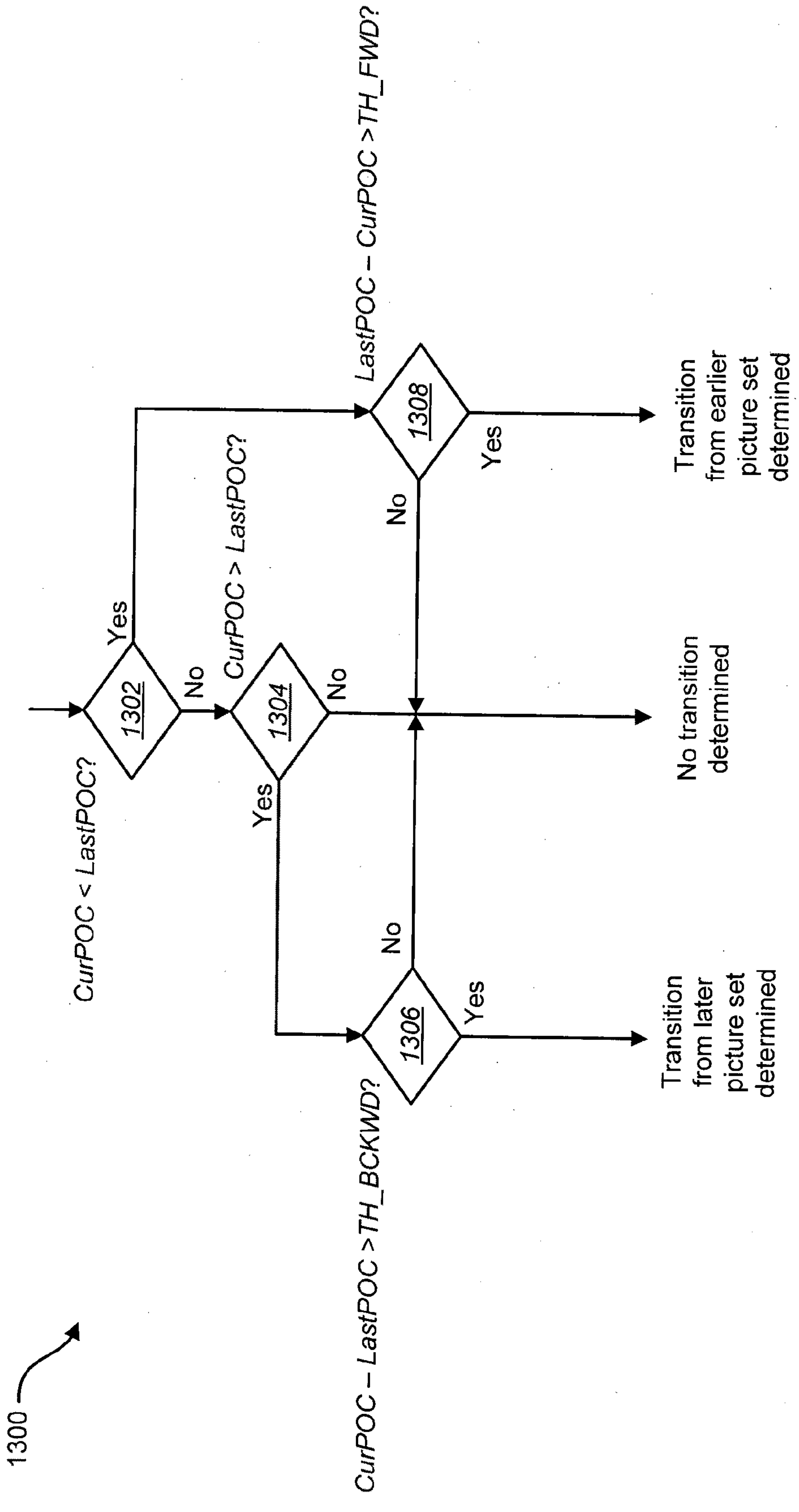

1300
CurPOC < LastPOC?
1302
Yes
No
CurPOC > LastPOC?
1304
No
Yes
LastPOC – CurPOC >TH_FWD?
1308
No
Yes
CurPOC – LastPOC >TH_BCKWD?
1306
No
Yes
Transition from earlier picture set determined
No transition determined
Transition from later picture set determined

FIG. 15

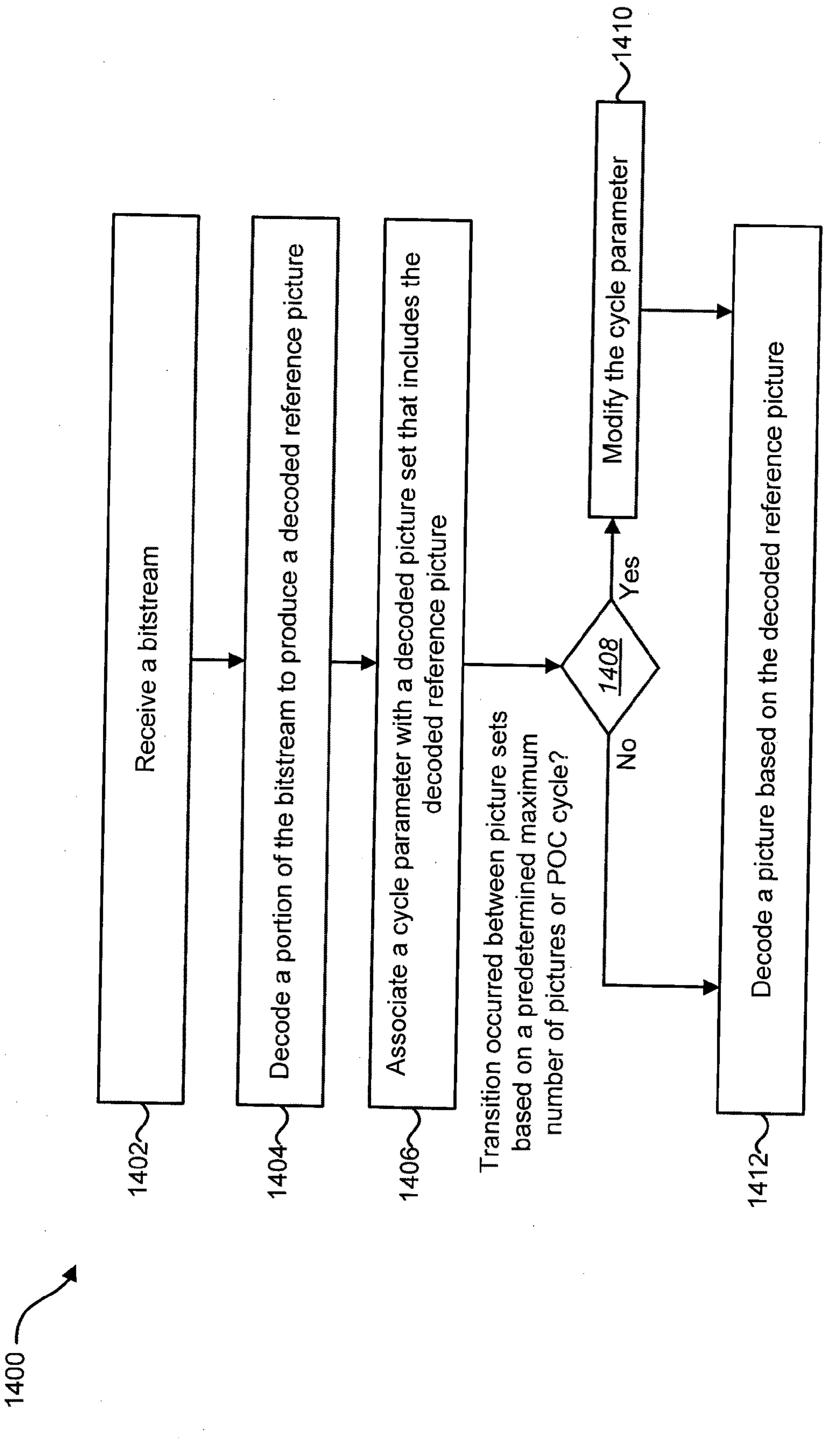

1400
Receive a bitstream
1402
Decode a portion of the bitstream to produce a decoded reference picture
1404
Associate a cycle parameter with a decoded picture set that includes the decoded reference picture
1406
Transition occurred between picture sets based on a predetermined maximum number of pictures or POC cycle?
1408
Yes
No
Modify the cycle parameter
1410
Decode a picture based on the decoded reference picture
1412

FIG. 16

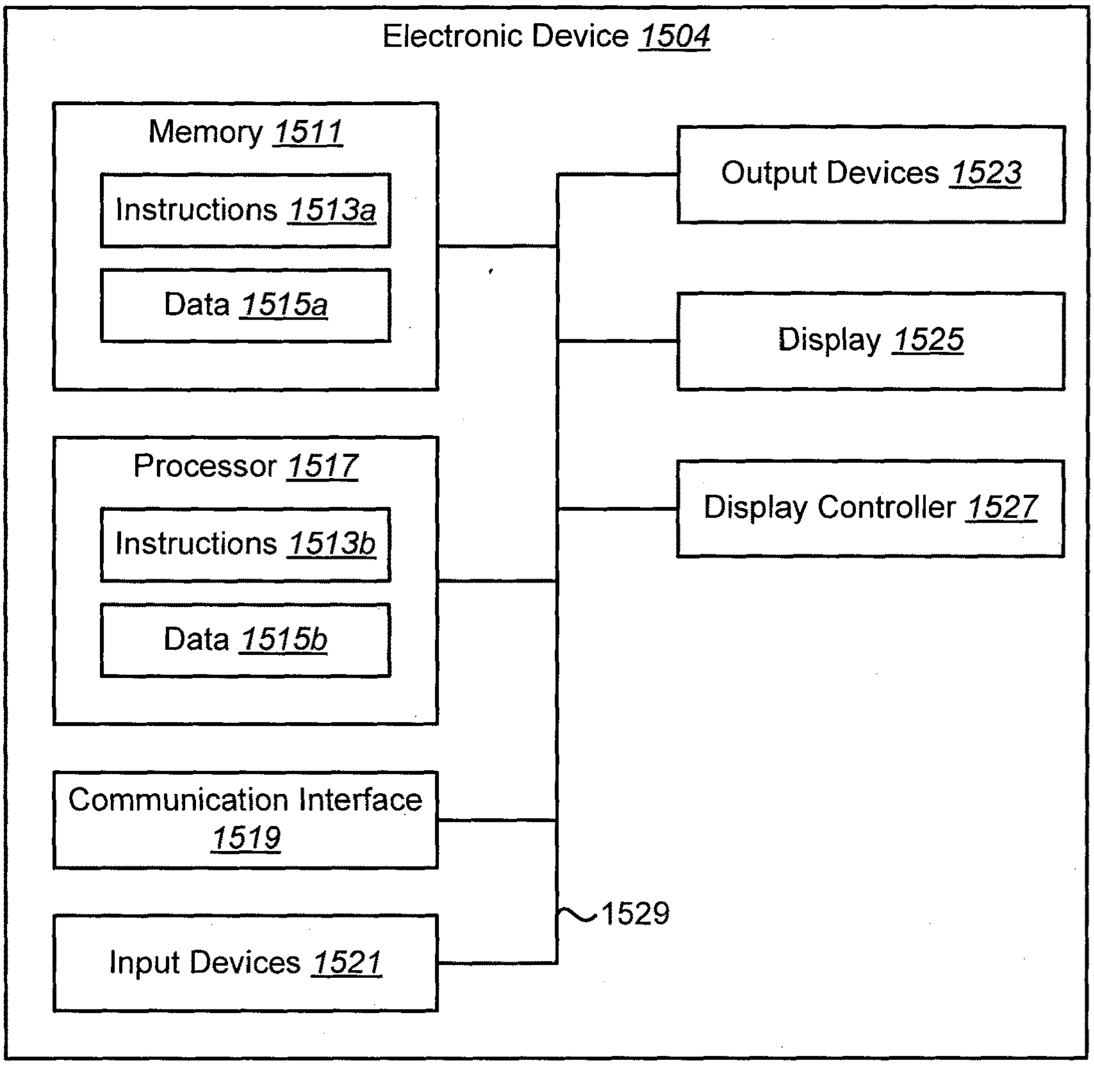
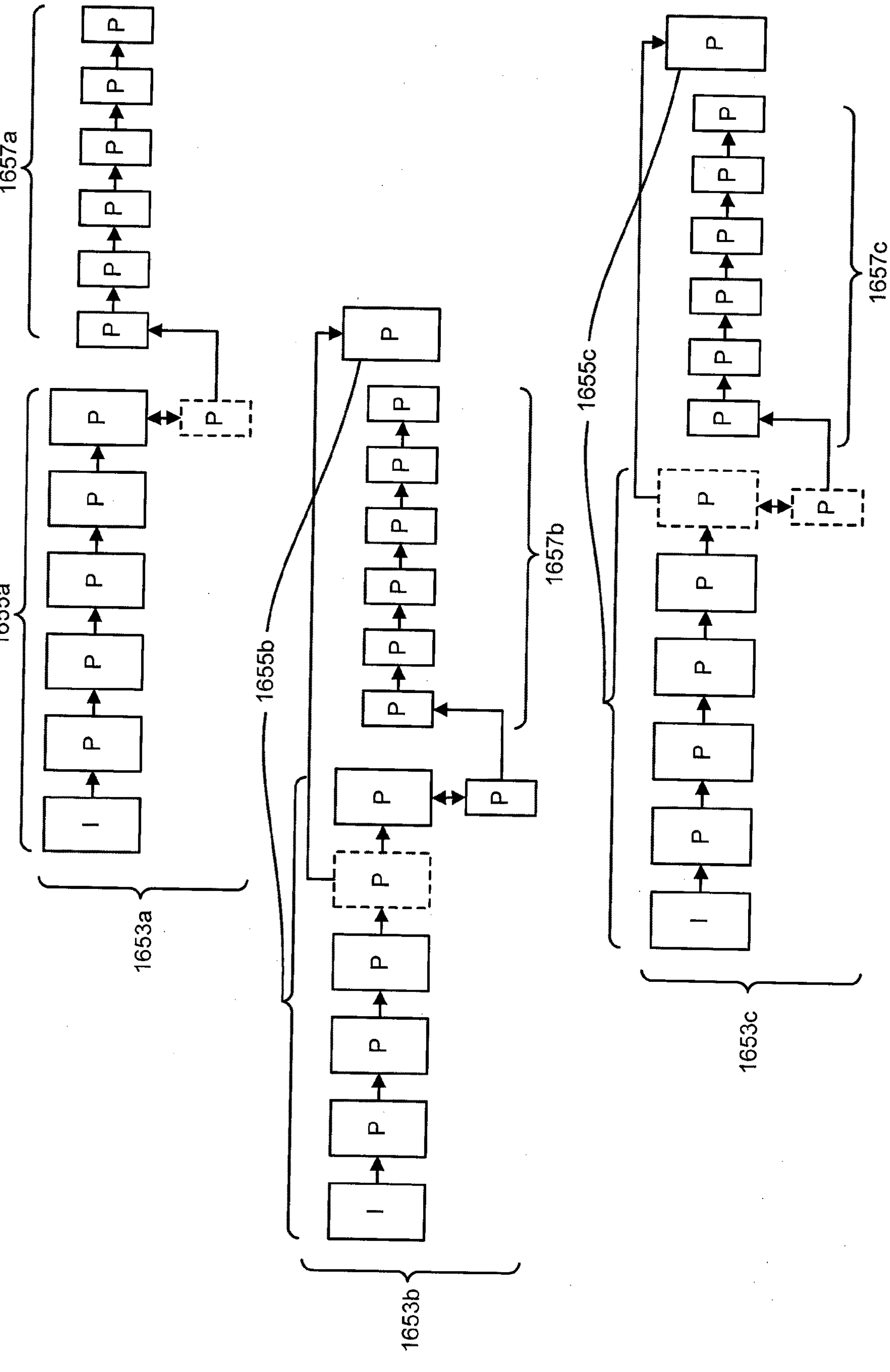

Electronic Device 1504
Memory 1511
Instructions 1513a
Data 1515a
Processor 1517
Instructions 1513b
Data 1515b
Communication Interface 1519
Input Devices 1521
Output Devices 1523
Display 1525
Display Controller 1527
1529

FIG. 17



Europäisches Patentamt
European Patent Office
Office européen des brevets

EUROPEAN SEARCH REPORT

Application Number
EP 22 16 5192

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
X	PARK Y ET AL: "Detection of CDR decoding status", 6. JCT-VC MEETING; 97. MPEG MEETING; 14-7-2011 - 22-7-2011; TORINO; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-F604, 2 July 2011 (2011-07-02), XP030009627, * the whole document * -----	1,2	INV. H04N19/423 H04N19/44 H04N19/573 H04N19/58 H04N19/70 H04N19/00
E	EP 2 727 342 A1 (ERICSSON TELEFON AB L M [SE]) 7 May 2014 (2014-05-07) * figure 15 * -----	1,2	
A	R Sjöberg ET AL: "JCTVC-F493: Proposed changes to the HEVC Working Draft", , 9 July 2011 (2011-07-09), pages 1-28, XP055054084, Retrieved from the Internet: URL:phenix.int-evry.fr/jct/index.php [retrieved on 2013-02-20] * the whole document * -----	1,2	TECHNICAL FIELDS SEARCHED (IPC) H04N
A	SJÖBERG R ET AL: "Absolute signaling of reference pictures", 6. JCT-VC MEETING; 97. MPEG MEETING; 14-7-2011 - 22-7-2011; TORINO; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-F493, 1 July 2011 (2011-07-01), XP030009516, * the whole document * ----- -/--	1,2	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
The Hague	13 May 2022	Martinière, Anthony

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons
....
& : member of the same patent family, corresponding document

Europäisches Patentamt
European Patent Office
Office européen des brevets

EUROPEAN SEARCH REPORT

Application Number
EP 22 16 5192

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
A,P	MISRA K ET AL: "AHG21: Long term picture referencing using wrapped POC", 7. JCT-VC MEETING; 98. MPEG MEETING; 21-11-2011 - 30-11-2011; GENEVA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-G713, 9 November 2011 (2011-11-09), XP030110697, * the whole document * -----	1,2	
A	EIFRIG: "POC Wrap, RAP, and Missing frames&fields", 6. JVT MEETING; 63. MPEG MEETING; 09-12-2002 - 13-12-2002; AWAJI, JP;(JOINT VIDEO TEAM OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16),, no. JVT-F098-L, 13 December 2002 (2002-12-13), XP030005663, ISSN: 0000-0428 * the whole document * -----	1,2	TECHNICAL FIELDS SEARCHED (IPC)
A	YING CHEN ET AL: "Comments on Clean Decoding Refresh Pictures", 5. JCT-VC MEETING; 96. MPEG MEETING; 16-3-2011 - 23-3-2011; GENEVA;(JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-E400, 11 March 2011 (2011-03-11), XP030008906, ISSN: 0000-0005 * the whole document * ----- -/--	1,2	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
The Hague	13 May 2022	Martinière, Anthony

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons
& : member of the same patent family, corresponding document

Europäisches Patentamt
European Patent Office
Office européen des brevets

EUROPEAN SEARCH REPORT

Application Number
EP 22 16 5192

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
A	WIEGAND T ET AL: "WD3: Working Draft 3 of High-Efficiency Video Coding", 20110330, no. JCTVC-E603, 30 March 2011 (2011-03-30) , XP030009014, ISSN: 0000-0003 * paragraph [7.4.1] * * paragraph [7.4.2.1] * * paragraph [7.4.3.3] * -----	1,2	
			TECHNICAL FIELDS SEARCHED (IPC)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
The Hague	13 May 2022	Martinière, Anthony

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons
....
& : member of the same patent family, corresponding document

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO.

EP 22 16 5192

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

13-05-2022

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 2727342 A1	07-05-2014	AP 3858 A	31-10-2016
		AU 2012276348 A1	16-01-2014
		BR 112013033191 A2	01-03-2017
		CA 2840349 A1	03-01-2013
		CA 2967452 A1	03-01-2013
		CN 103797797 A	14-05-2014
		DK 2727342 T3	03-10-2016
		DK 3091744 T3	02-10-2017
		DK 3229474 T3	11-03-2019
		DK 3410725 T3	20-09-2021
		EP 2727342 A1	07-05-2014
		EP 3091744 A1	09-11-2016
		EP 3229474 A1	11-10-2017
		EP 3410725 A1	05-12-2018
		EP 3902258 A1	27-10-2021
		ES 2595732 T3	03-01-2017
		ES 2645943 T3	11-12-2017
		ES 2714756 T3	29-05-2019
		ES 2892283 T3	03-02-2022
		HK 1192390 A1	15-08-2014
		HU E028853 T2	30-01-2017
		IL 230146 A	31-07-2016
		JP 5616562 B2	29-10-2014
		JP 5889982 B2	22-03-2016
		JP 5889983 B2	22-03-2016
		JP 6148293 B2	14-06-2017
		JP 2014523690 A	11-09-2014
		JP 2015015755 A	22-01-2015
		JP 2015015756 A	22-01-2015
		JP 2015222971 A	10-12-2015
		KR 20140053117 A	07-05-2014
		KR 20150043559 A	22-04-2015
		KR 20150046359 A	29-04-2015
		MA 35202 B1	02-06-2014
		MY 166340 A	25-06-2018
		PL 2727342 T3	31-01-2017
		PL 3091744 T3	31-01-2018
		PL 3229474 T3	31-05-2019
		PL 3410725 T3	21-02-2022
		PT 2727342 T	01-08-2016
		PT 3091744 T	31-08-2017
		PT 3229474 T	04-02-2019
		PT 3410725 T	06-09-2021
		RU 2014102959 A	10-08-2015
		RU 2016125119 A	04-12-2018
		RU 2019137954 A	25-05-2021

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO.

EP 22 16 5192

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

13-05-2022

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		SG 10201605216X A	30-08-2016
		TR 201901275 T4	21-02-2019
		US 2013215975 A1	22-08-2013
		US 2017041633 A1	09-02-2017
		US 2018035131 A1	01-02-2018
		US 2018338157 A1	22-11-2018
		US 2019297346 A1	26-09-2019
		US 2020288167 A1	10-09-2020
		WO 2013002700 A1	03-01-2013
		ZA 201309396 B	30-09-2015
		ZA 201500885 B	26-10-2016

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 29196111 **[0001]**
- US 28701511 **[0001]**
- US 27319111 **[0001]**